ABSTRACT

HENDERSON, NATHAN LEE. Deep Learning-Based Multimodal Affect Detection for Adaptive Learning Environments. (Under the direction of Dr. James C. Lester).

As students engage with digital learning environments, they may experience a wide range of emotions or affective states throughout the learning process. Consequently, it is desirable for adaptive learning environments to encourage students to remain in states of affect that promote increased learning and engagement, and to simultaneously discourage students from entering or remaining in affective states that are correlated with disengagement or decreased learning. A core component of affective computing is the ability to accurately predict and recognize affective states based on a person's behavioral cues, and this capability is critical for enabling user-sensitive mechanisms that dynamically detect and respond to instances of affect. Such mechanisms show significant promise for integration into adaptive learning environments for the purpose of promoting emotion regulation in students and providing an enhanced learning experience.

Affect detection systems have begun to simultaneously integrate combinations of sensor-based and interaction-based modalities, and these multimodal systems have shown significant promise in terms of predictive performance compared to unimodal baselines. However, these systems frequently encounter problems that inhibit data capture and result in sparse multimodal datasets, such as sensor failure, mistracking, and noise. Additional questions arise pertaining to the optimal method of combining multiple modalities, the most effective modeling approach in terms of predictive capacity, and the generalizability of affective computing approaches across different modalities, learning environments, and populations. In recent years, deep learning has become an integral part of machine learning research, due to its ability to effectively model non-

linear functions within high-dimensionality data; however, its effectiveness for multimodal affect detection tasks is still an area of ongoing research.

This dissertation presents a multimodal framework that utilizes a series of deep learning-based approaches to enhance affect detection for two different student populations that are engaged with two digital learning environments. The dissertation introduces an evaluation of several standard machine learning models in addition to deep neural networks as a solution to predicting annotated instances of student affect using different combinations of interaction-based and sensor-based modalities. In addition, this dissertation investigates the impact that various forms of feature-level and decision-level multimodal data fusion have on the framework's predictive accuracy. The dissertation also addresses the issue of data sparsity through two separate deep learning-based approaches: data imputation and data augmentation. Finally, the dissertation investigates increasingly complex deep learning approaches as a means to improve the predictive performance of the affect models, using methods such as multi-task learning, affective dynamics modeling, and cross-stitch networks.

The preliminary research presented in this dissertation was conducted using multimodal data captured from students engaged with a game-based learning environment for teaching emergency medical care, with results indicating that deep learning-based approaches to affect modeling, data imputation, and data augmentation outperform non-neural baseline models. Following this work, the generalizability and effectiveness of the multimodal framework was verified using a separate dataset consisting of multimodal data captured from undergraduate students engaging with a tutor-driven programming environment. In this investigation, students self-report affect across an entire learning session instead of affect being captured through real-field observations. As a result, the multimodal framework is evaluated across different student

populations, modalities, data representations, learning environments, and affect annotation schemes. Results indicate that deep learning yields improved performance for different components of multimodal affective computing tasks, and the techniques applied in this dissertation generalize across different learning environments, modalities, and student populations. Future implications of the demonstrated multimodal affect detection framework include the enabling of learner-sensitive intervention mechanisms within future adaptive learning environments, leading to enhanced student engagement and improved learning outcomes.

Deep Learning-Based Multimodal Affect Detection for Adaptive Learning Environments

by
Nathan Lee Henderson

A dissertation submitted to the Graduate Faculty of
North Carolina State University
in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy

Computer Science

Raleigh, North Carolina
2022

APPROVED BY:

_____	_____
Dr. James C. Lester Committee Chair	Dr. Tiffany Barnes
_____	_____
Dr. Min Chi	Dr. Jing Feng

DEDICATION

Dedicated to my wife Ashley, and my dog Marshall for their endless love and support.

BIOGRAPHY

Nathan Lee Henderson was born in Seoul, South Korea on August 9, 1994, but was raised in Huntsville, Alabama. He completed a Bachelor of Science in Electrical and Computer Engineering from Auburn University in 2016. During this time, he had the opportunity to work for three summers as a software engineering intern at Science Applications International Corporation, a contractor for the U.S. Department of Defense. In addition to this experience, he also worked as an undergraduate research assistant for the Auburn Cyber Research Center.

After completing his undergraduate education, Nathan returned to Huntsville, AL, and worked as a software engineer for another Department of Defense contractor, Radiance Technologies, prior to beginning his Master of Science in Computer Science at The University of Alabama in Huntsville (UAH). During his time at UAH, he worked as a Graduate Research Assistant for the Information Technology and Science Center. This experience first introduced Nathan to the world of Artificial Intelligence and Machine Learning (AI/ML), and led him to complete his Master's thesis on deep learning approaches to human action recognition. Upon completing his Master's degree in 2017, he worked on a number of AI/ML tasks as a Data Scientist for Corvid Technologies, another Department of Defense contractor in Huntsville.

In 2018, Nathan relocated to Raleigh, NC, to begin his Ph.D. in Computer Science at North Carolina State University where he was a Graduate Research Assistant for the IntelliMedia Lab under the supervision of Dr. James Lester. His doctoral research at NC State primarily focused on deep learning applications for student modeling, including affective computing, multimodal interaction, natural language processing, and learning outcome prediction. After completing his Ph.D., Nathan will return to Radiance Technologies where he will work as an AI Research Scientist.

ACKNOWLEDGMENTS

First and foremost, I would like my advisor Dr. James Lester for his endless support and guidance throughout my Ph.D. studies. Through his advice and encouragement, I have become a better researcher, thinker, and learner. His example of scholarly professionalism is truly admirable. In addition, I would like to thank the other members of my doctoral committee, Drs. Tiffany Barnes, Min Chi, and Jing Feng, for their insightful feedback and advice while completing this dissertation.

I would like to thank Jon Rowe for providing invaluable mentorship and guidance, answering my endless questions about various research topics, and being a wonderful source of advice for my academic pursuits as well as daily life. I would also like to thank Wookhee Min for his insightful advice and technical expertise, which has contributed to a vast majority of my doctoral research. In addition, I would like to thank Rob Taylor and Barry Liu for frequently taking the time to answer all of my software development questions and for providing support in various ways across my different research projects.

I extend special thanks to the other members of the IntelliMedia group for their support and encouragement over the years. These colleagues were outstanding to work with and made my time as a Ph.D. student productive but also fun and enjoyable. In particular, I would like to thank Dan Carpenter and Andrew Emerson, who are great colleagues but even better friends.

There is a long list of collaborators to whom I owe special thanks for their support and input on many different research projects, including (but certainly not limited to): Halim Acosta, Ryan Baker, Keith Brawner, Fahmid Morshed Fahid, Alex Goslen, Stephen Hutt, Vikram Kumaran, Seung Lee, James Minogue, Bradford Mott, Jaclyn Ocumpaugh, Jay Pande, Luc Paquette, Kyungjin Park, Emily Tracy, Eric Wiebe, and Ziwei Wu.

Finally, I would like to thank all of my friends and family who have supported me relentlessly throughout this journey; their thoughts, prayers, and well-wishes were truly felt. In particular, I would like to thank my parents, John and Teresa Henderson, for their unending love and support throughout my life. Without them, I would not be the person I am today. Finally, I would like to thank my wife, Ashley Henderson, for her constant love, support, and encouragement as I have pursued a Ph.D. She has graciously navigated the late nights, early mornings, traveling, and general stress that comes with graduate school, and has done so while continuing to encourage and believe in me while also advancing her own professional career. Without her support, none of this would have been possible.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

Affect plays a key role in student learning (D'Mello, 2013). Students experience a diverse range of affective experiences in different learning environments, and these are influential in shaping students' cognitive and motivational processes (Loderer, Pekrun, & Lester, 2018). For example, positive affective states, such as *flow* or *delight*, are often associated with engaged learning (Craig et al., 2004), whereas other states, such as *frustration* or *confusion*, have complex relationships with learning. Student frustration can lead to boredom and disengagement, but it may also signify a student's concerted effort toward overcoming an impasse or challenge, which is an important component of the learning process (Pardos et al., 2014). The ability to accurately recognize students' affective states is critical for enabling adaptive learning technologies that provide affect-sensitive support to promote engagement, facilitate emotion regulation, and enhance student learning outcomes (Cooper, Arroyo, & Woolf, 2011; DeFalco et al., 2018; Grafsgaard et al., 2014).

Recent years have seen growing interest in the use of multimodal systems to capture and recognize students' affective states (Bosch et al., 2016; Wu et al., 2019; Yang et al., 2018). A broad range of *sensor-based* modalities have been used to model student affect in adaptive learning environments, including facial expression (Bosch et al., 2016), eye gaze (Rajendran et al., 2018), electrodermal activity (EDA) (Vail et al., 2016), and electroencephalography (EEG) (Soleymani et al., 2016). An important benefit of sensor-based affective modeling is the potential for generalizability across different learning environments; sensor-based affect detectors do not necessarily rely on specialized representations from a particular learning environment or educational subject. Furthermore, multimodal sensor systems can be assembled using low-cost

commodity hardware, reducing barriers to creating affect-sensitive learning technologies that can be deployed across a range of educational settings.

An alternative to sensor-based affect detection is utilizing interaction data to induce *sensor-free* affect detectors, which can be used in contexts where it may be difficult or prohibitive to use physical hardware sensors (Botelho, Baker, & Heffernan, 2017). Sensor-free features are typically derived from trace log data generated by learner interactions with digital learning environments (Jiang et al., 2018). Notably, sensor-free affect detectors typically avoid challenges inherent in the use of physical sensors, including calibration issues, hardware failure, and mistracking (Baker et al., 2012).

Several studies have suggested that multimodal affect detection improves model accuracy and robustness compared to unimodal techniques (D'Mello & Kory, 2012; Grafsgaard et al., 2014). Similar to the human perception of emotion, which is multi-sensory, multimodal sensor systems produce a rich, multivariate set of indicators of affect, which can be modeled using multimodal machine learning techniques (Baltrušaitis, Ahuja, & Morency, 2019). Multimodal sensor-based affect detection has demonstrated significant promise in predictive tasks related to self-reported affect data (Arroyo et al., 2009) and normalized learning gain (Grafsgaard et al., 2014), and has been deployed in a number of learning environments such as classroom (Arroyo et al., 2009) and laboratory settings (Chan, Ochoa, & Clarke, 2020). However, multimodal sensor systems raise important challenges (DeFalco et al., 2018). Devising effective representations of multimodal sensor data is critical to the effectiveness of computational models of learner affect (Grafsgaard et al., 2014; Soleymani et al., 2016; Yang et al., 2018). Additionally, sensors can produce noisy data, suffer from calibration and mistracking issues, or run into data storage constraints, resulting in unreliable data or data loss (Henderson, Emerson, et al., 2019). Investigating methods to enhance

the resilience of multimodal affect detectors in the face of noisy or missing sensor data is a key challenge.

Deep learning architectures have seen a significant increase in usage across various domains and tasks due to their capability of solving computationally complex problems using high-dimensionality feature extraction. With increased availability of computing resources, the training of deep learning models has become less time-consuming and more generalizable across various domains. Deep learning has demonstrated significant promise in different research fields such as data translation (Isola et al., 2017), augmentation (Chatziagapi et al., 2019), imputation (Yoon, Jordon, & Schaar, 2018), and natural language processing (Young et al., 2018). In addition, a significant amount of research has been devoted to deep learning within multimodal frameworks, with results often yielding improved results compared to non-deep learning methods or unimodal models (He et al., 2015; Henderson, Rowe, et al., 2020; Wu et al., 2019; Yang et al., 2018).

As noted above, both sensor-based and interaction-based affect detection systems often encounter issues that distort or prohibit consistent data capture. Physical and physiological sensors can be impeded by noise (Henderson, Emerson, et al., 2019), mistracking (Bosch, Chen, et al., 2015), and data storage constraints. Interaction log-based modalities also suffer from issues such as software or hardware failure, network connection problems, data logging and transfer problems (Spain et al., 2019), and incompleteness issues (Sabourin et al., 2013). Data loss can also occur due to practical challenges that are common in educational settings, including schools' reliance on aging computers, accidental unplugging, and student mishandling of machines. Common approaches to addressing these challenges include discarding data samples with missing data and simple imputation methods such as mean imputation. However, discarding data significantly reduces the amount of training data available for machine learning models for affect detection.

Therefore, evaluating more intelligent data imputation approaches such a deep learning shows promise for improving the predictive performance of multimodal affect detection models.

A promising approach to affect detection is predicting multiple affective states with a single output vector. A static output vector can represent a single affective sequence consisting of multiple target variables, each representing an affective state at a particular time interval. Alternatively, another example of multi-objective affect detection is the simultaneous prediction of multiple affective states (e.g., a student self-reporting multiple affective states across a single learning session). Because these approaches necessitate a single model making multiple concurrent predictions, multi-task learning (MTL) provides a natural solution. MTL has several advantages over single-task modeling, including the ability to share feature representations and learned weights across multiple target variables, which introduces a form of model regularization (Zhang & Yang, 2017). Multi-task models also require a significantly lower number of parameters compared to the total number of parameters required by separately trained models for each individual task, while also allowing the model to inherently learn the interwoven relationships between the target variables (Shui et al., 2019). Prior work indicates that MTL outperforms single-task learning in terms of predictive performance for a variety of tasks (Geden et al., 2020; Long et al., 2017). However, the use of multiple tasks poses challenges regarding the weighting of each task's predictive performance during training, as different predictive tasks often vary in nature and intended purpose. Additionally, the appropriate balance of task-specific and shared latent representations within a multi-task model can vary as well and have a noticeable impact on model performance.

This dissertation investigates the effectiveness of various deep learning techniques within a multimodal affect detection framework utilizing a combination of sensor-free and sensor-based

data channels. The framework is initially evaluated using gameplay, posture, and EDA data captured from students engaged with a game-based learning environment for teaching emergency medical care. The predictive performance of deep learning for recognizing different affective states is evaluated using two modality combinations (posture + EDA, posture + interaction), and the effectiveness of different methods of multimodal data fusion is also investigated. Additionally, the dissertation explores deep learning-based approaches to addressing two critical issues within sensor-based multimodal affect detection: missing multimodal data imputation, and resolution of class imbalances. Finally, the dissertation explores the use of affect sequences as auxiliary multi-task target variables as a means to improve the predictive performance of the affect models through the use of different MTL deep learning architectures. The deep learning models are evaluated against non-neural baseline models using *k*-fold cross-validation and multiple performance evaluation metrics.

The dissertation also investigates the generalizability of the affect detection framework across multiple learning environments. For this evaluation, we use another multimodal affect-oriented dataset captured from undergraduates learning Java programming through a tutor-driven development environment. The modalities in this dataset consist of interaction-based features based on students' actions within the environment, as well as facial expression data captured from a front-facing webcam. Additionally, text-based chat dialogue between the student and tutor during the learning sessions is retained to form a discourse-based modality. As a result, this dissertation also investigates different word embedding representations of student and tutor dialogue and evaluates their respective impact on the multimodal affect models' predictive performance. Following each learning session, students self-reported levels of *engagement* and *frustration*. The values from the post-test surveys are used as the target variables for the multimodal affect detection

tasks for this particular student population. This dataset allows us to evaluate the predictive performance of the deep learning approaches noted above using a different learning environment, student population, modality set, and affective annotation scheme compared to the initial experiments mentioned in the previous paragraph. Additionally, the generalizability of the affect detection framework is evaluated in terms of multimodal data imputation performance, while also investigating the impact of the imputation on the predictive performance of the affect models. Finally, the dissertation investigates the effectiveness of multi-task modeling techniques to enhance the affect detection framework by predicting each of the affective states simultaneously. This approach is taken as an alternative MTL technique compared to the affect sequence modeling approach due to the fact that only a single affect label exists for each session, meaning that affect sequences are not available for this particular dataset. As a result, this dissertation yields a generalizable multimodal affect detection framework that utilizes sophisticated deep learning techniques for modeling and predicting instances of affect based on student behavior, accurately imputing missing or otherwise invalid multimodal data samples, and enhancing the affect models' performance through generative data augmentation for resolving class imbalances within the data corpora.

1.1 Thesis Statement and Hypotheses

The high-level objective of this dissertation is to present a multimodal affect detection framework that utilizes deep learning architectures within the different components of the framework, with the primary purpose of enhancing the predictive performance of the affect models. The relationship between multiple modalities is often complex, and deep learning provides an effective approach to modeling such high-dimensionality, non-linear functions as they relate to student affect. The optimal method of effectively combining multiple modalities while retaining each modality's

predictive value is an area of ongoing research and is a topic of investigation within this dissertation.

Obtaining data samples with a corresponding affect label is often logistically complex and can be limited by a number of factors. The amount of available data can be limited by the affect annotation method as well. For example, labeling multiple instances of affect within a single student's interaction with a learning platform typically produces much more data than administering a self-report survey at the end of each student's learning session. Operational issues that occur with the use of sensor-based modalities or even sensor-free modalities are fairly common and can lead to significant data loss. To address these factors, the multimodal affect detection framework implements deep learning-based methods of addressing potential issues in the model training process that can be attributed to imbalanced or sparse data (data augmentation) as well as missing or invalid data (data imputation).

Affect detection models are often deployed as binary classification models or are otherwise implemented within single-task prediction frameworks. However, multi-task learning has seen an increased focus in recent years due to its capability to simultaneously model multiple functions using shared feature representations. This functionality serves as a form of model naturalization, which has shown to reap benefits in terms of predictive model accuracy in prior student modeling tasks (Geden et al., 2020). Environments where multiple affect annotations are captured during a single learning session or where the presence of multiple concurrent affective states are considered provide potential avenues for the implementation and evaluation of multi-task affect detection models, and these are also investigated in this dissertation.

Multimodal affect detection systems utilize a wide variety of modalities, learning platforms, affect annotation schemes, and student populations. To verify the generalizability of

our multimodal affect detection framework, this dissertation investigates the effectiveness of the deep learning-based components using two different multimodal datasets centered around student affect prediction tasks. The deep learning-based components are evaluated in terms of predictive performance across multiple modality combinations and different learning platforms.

Deep learning has been demonstrated to achieve higher predictive performance of various states of affect compared to a number of traditional methods, including support vector machines, random forest, Naïve Bayes, linear regression, and decision trees (He et al., 2015; Henderson, Rowe, et al., 2020; Kratzwald et al., 2018). However, a significant portion of multimodal affective computing does not consider deep learning techniques, which limits the evaluation of the predictive capabilities of various multimodal affect models. As a result, a prudent avenue of investigation is to perform deep learning-based affect detection with different combinations of sensor-based and sensor-free modalities, and to compare the deep learning-based approaches against classical machine learning models commonly found in the literature. Additionally, the method of combining modalities can be directly correlated with the affect models' predictive performance, as the multimodal data fusion determines the representation and dimensionality of the data. This dissertation explores the predictive performance of deep learning-based models for multimodal affect detection, using multiple modality combinations in addition to three variations of feature-level and decision-level data fusion.

A common justification behind the exclusion of deep learning within affective computing tasks is the relatively small size of many multimodal datasets. To enable the use of deep learning on sparse datasets, additional work within this dissertation explores two methods of dataset expansion: data augmentation, and data imputation. The data augmentation work focuses on resolving class imbalances through the use of generative models for producing synthetic

multimodal data across all modalities, for the purpose of providing additional training samples for affective states that are infrequent or uncommon in naturalistic settings (such as *surprise* or *frustration*). The primary objective of the data imputation work is to accurately approximate missing or invalid values within different modalities, while minimizing potential impact on the training and predictive capacity of student affect models.

Finally, multi-task learning has been shown in prior work to provide improved predictive performance compared to single-task modeling approaches. Affect detection has the potential to harness this capability in situations where a student is experiencing multiple emotions at one time, if a student experiences multiple emotions throughout a single learning session, or if a student self-reports multiple affective states across the entire session. The effectiveness of deep learning-based multi-task modeling is investigated across the two data corpora and is evaluated against both neural and non-neural single-task and multi-task baselines.

Two neural approaches to data imputation are investigated: denoising multimodal autoencoders and generative modeling techniques. These approaches explore the benefit to imputing a single modality suffering from missing or invalid data points by using a corresponding modality. This dissertation investigates the performance of the data imputation in terms of imputation performance, and in terms of the impact the imputation has on pre-trained affect detection models. The imputation and subsequent inclusion of additional multimodal data can potentially lead to decreased performance on pre-trained affect models if the imputation is not sufficiently accurate. Consequently, the data imputation is evaluated from this perspective to investigate the tradeoff between data availability and imputation accuracy against the predictive accuracy of the affect models. Preliminary work has demonstrated improved performance in terms of imputation accuracy and impact on affect models on a single data corpus using deep learning-

based architectures; but these methods are evaluated across multiple data corpora to validate the generalizability of the imputation methods.

The multimodal affect detection framework should be generalizable across different learning environments, modalities, and populations. Preliminary work has investigated the effectiveness of the various components of the framework, using a single data corpora captured from military cadets interacting with a game-based learning environment designed to provide training in properly administering emergency medical care. However, to verify the generalizability of the framework, the deep learning approaches are also empirically evaluated on an additional data corpus that differs from the preliminary work in terms of available modalities, student demographics, learning platform, and affect annotation method. In addition to the evaluation of the predictive models across multiple datasets, the deep learning-based approaches to data augmentation and imputation and the variants of multimodal data fusion are evaluated as well. The deep learning-based approaches with both data corpora are evaluated comparatively alongside non-neural baseline models commonly used within affect detection literature. Because one of the data corpora consists of text-based chat dialogue, this dissertation also investigates different word embedding techniques (e.g., BERT, GloVe, Word2Vec, etc.) to determine which embedding model induces the highest predictive performance from the affect model, and also explores whether separating the dialogue based on source into separate embeddings improves the models' accuracy. This dissertation also investigates the impact of different dialogue patterns on the models' predictive accuracy across a single learning session.

The resulting dissertation research yields a multi-faceted multimodal affect detection pipeline that demonstrates enhanced predictive performance across different combinations of input data channels. The pipeline addresses common problems within multimodal affective computing,

namely data imbalance and sparseness, in addition to missing or invalid data points within particular modalities. It is anticipated that this proposed pipeline improves and advances various components of multimodal affect detection.

The following hypotheses are tested in several experiments that evaluate multimodal affect detection models devised using deep learning techniques. The deep learning techniques consist of both generative and predictive modeling approaches and are implemented using two data corpora generated from two distinct learning platforms and data collections.

- Hypothesis 1: Deep learning techniques achieve higher performance of student affect in terms of predictive accuracy and data augmentation impact when compared to standard non-deep learning classification models and upsampling techniques. Deep learning multi-task models trained using a student's affect sequence data outperform single-task deep learning models and non-deep learning models in terms of predictive accuracy.
- Hypothesis 2: Deep learning imputation models such as conditional generative adversarial networks or stacked denoising autoencoders trained on multimodal data outperform standard non-deep learning imputation methods while having minimal adverse impact on student affect models and improving student affect model performance in terms of predictive accuracy.
- Hypothesis 3: Deep learning techniques that enhance multimodal affect detection are generalizable across two different learning environments, affect annotations, and modalities. These techniques are effective when applied within two distinct data corpora.

1.2 Contributions

The research presented in this dissertation has been informed by analytical, empirical, and computational investigations of deep learning-based models of student affect in addition to applications of neural approaches for addressing sparsity and imbalances within the multimodal

datasets. We aim to design a multimodal affect detection framework that is robust to common issues with multimodal data capture, while retaining generalizability across different domains and platforms. Based on the experimental results, the work that is described in this dissertation makes the following contributions:

- A novel multimodal affect detection framework that implements various deep learning-based technologies for tasks related to data sparsity and affect modeling. This includes the evaluation of feedforward neural networks (FFNNs) as an alternative approach to affect detection and the use of generative models such as auxiliary classifier generative adversarial networks as a means of augmenting multimodal data used to induce the predictive models (Henderson, Min, et al., 2020b; Henderson, Rowe, et al., 2019, 2020).
- An empirical evaluation of different deep learning-based approaches to missing data imputation across various modalities. This includes investigating stacked denoising autoencoders as a means of imputation using two sensor-based modalities (Henderson, Emerson, et al., 2019) and a combination of sensor-based and interaction-based modalities, and conditional generative models including generative adversarial networks and variational autoencoders using interaction-based and sensor-based modalities (Henderson, Min, et al., 2020a).
- An empirical investigation of the effectiveness of deep learning-based multi-task affect detection models. This work includes an evaluation of multi-task models such as feedforward neural networks and cross-stitch networks, as a means to evaluate the balance between task-specific and shared parameters within each model (Henderson, Min, et al., 2021b).
- An evaluation of the generalizability of the deep learning-based multimodal affect detection framework in terms of predictive performance. The affect detection framework produces high

predictive performance across two distinct learning environments, demonstrating promise as a student affect modeling approach that is scalable to unseen learning environments.

- An evaluation of the generalizability of the data imputation capabilities of the deep learning-based multimodal framework. This includes evaluating the techniques for accurate imputation performance while simultaneously minimizing the impact on the predictive performance of the affect detection models. Evaluating the multimodal data imputation techniques across different learning environments and modalities provides additional evidence for the scalability of the affect detection framework.

1.3 Organization

The remainder of this dissertation is as follows. Chapter 2 provides a background on affective computing, particularly multimodal affect detection, and recent work on different deep learning applications. Chapter 3 provides a brief overview of the overall body of work and the various components of the deep learning-based multimodal affect detection framework. Chapter 4 describes the multimodal affect detection data corpora that are examined in the dissertation, in addition to the feature engineering processes for the individual data channels. Chapter 5 describes preliminary research into the predictive value of different word embedding representations of a dialogue-based modality in one of the multimodal datasets used in this dissertation. Chapter 6 describes preliminary work exploring the impact of multimodal data fusion and a multimodal processing pipeline for posture and electrodermal activity contained in one of the multimodal datasets. Chapter 7 describes an autoencoder-based approach to imputing missing values in the electrodermal activity modality and its impact on the multimodal affect models' predictive performance across both data corpora. Chapter 8 investigates further the multimodal data fusion across different sensor-based and sensor-free modalities for the two multimodal corpora. Chapter

9 describes a deep learning approach for data augmentation to improve the predictive performance of the affect models. Chapter 10 explores a generative deep learning approach for imputing artificially-masked values across different modalities within the two datasets. Chapter 11 investigates the use of multi-task learning and cross-stitch networks as an alternative method to improve predictive performance of the different affective states between the two datasets. Chapter 12 revisits the hypotheses presented in Chapter 1, summarizes the contributions of this dissertation, and provides different directions for future research related to this work.

CHAPTER 2

BACKGROUND AND RELATED WORK

The proposed affect detection framework is situated at the intersection of two primary research fields: 1) multimodal interaction, which involves the use of multiple data channels (i.e., modalities) that capture different dimensionalities of a user's physical behavior, and 2) deep learning, a subset of machine learning that utilizes multi-layered artificial neural networks combined with non-linear activations, allowing a neural network to approximate virtually any continuous function. Because of this capability, deep learning has seen a vast increase in usage within affective computing tasks, particular within educational contexts and environments. A significant portion of deep learning research within education investigates neural networks as effective models for affect detection tasks. However, deep learning research that explores tasks such as missing data imputation or data augmentation is relatively underemphasized within the education domain. This chapter is organized in the following manner: Section 2.1 describes prior work that addresses the issue of recognizing or detecting instances of affect. Section 2.2 introduces research that explores the effectiveness of multimodal approaches to various user modeling tasks and various techniques of integrating multiple modalities into a single cohesive model. Section 2.3 provides a brief history of deep learning and introduces the neural network architectures that are used in the preliminary work described in later chapters.

2.1 Affective Computing

Affective computing has received growing interest from a number of fields including computer science, psychology, physiology, and cognitive science (Picard, 2010; Tao & Tan, 2005). This research field encompasses a variety of tasks, such as sentiment analysis (You et al., 2015), human-

computer interaction (Picard, 1999; Zeng et al., 2009), and affect detection (Baker et al., 2012; Bosch, D'Mello, et al., 2015; D'Mello & Kory, 2015). The capability to accurately detect instances of affect can potentially allow intelligent systems to respond and adapt to particular affective states to increase the system's effectiveness (Brave & Nass, 2009). In particular, affect detection is of utmost importance within the education domain as emotion and affect are integral to components of a student's behavior such as decision making (Kratzwald et al., 2018; Picard, 2000), cognitive processes (Harley et al., 2015), self-expression (AlZoubi, D'Mello, & Calvo, 2012; Wei, Jia, & Chen, 2016), and learning gain (Grafsgaard et al., 2013b). Predictive models of student affect can be integrated into adaptive learning environments to enable various mechanisms such as dynamic feedback (DeFalco et al., 2018), expressive virtual agents (Cabada et al., 2015; Meudt et al., 2016), and hint generation (Tiam-Lee & Sumi, 2018).

There are multiple approaches to capturing instances of student affect, including real-time field observations (DeFalco et al., 2018), post-hoc third-party analysis (Soleymani et al., 2016), intentional displays of affect (Psaltis et al., 2016), and self-reporting (Grafsgaard et al., 2014; Soleymani et al., 2016). However, each method of capturing and annotating instances of student affect contains inherent issues that should be considered when determining the practicality of affect detection models. For example, intentional displays of affect (i.e., requesting that a person act out a display of "anger") may often result in capturing unnaturalistic or exaggerated behavioral pattens, which can lead to inherent bias or desensitization in the affect models. Additionally, post-hoc analysis to determine instances of affect necessitates a data-rich capture of student behavior, often through the use of video or audio recordings (Soleymani et al., 2016). However, this can raise privacy and ethical concerns and is impractical in environments or situations where such forms of data capture are prohibitive. Self-reporting is a common method of capturing instances

of student affect (AlZoubi et al., 2020; Chatziagapi et al., 2019; Grafsgaard et al., 2012; Thomas, Nair, & Jayagopi, 2018; Tiam-Lee & Sumi, 2018; Wu et al., 2019). However, frequent self-report surveys can be intrusive and cause disruption in the learning process. Additionally, self-report surveys are reliant on reinterpretation and recall, and require a certain level of self-awareness (Ocumpaugh, Baker, & Rodrigo, 2015). Kassam & Mendes have demonstrated that the actual act of labeling an emotional state can have a direct impact on physiological responses (2013), which can lead to noise within sensor-based affect detection frameworks. Finally, self-presentation remains an issue within self-report surveys (e.g., a student reporting that they are focused when in reality they are bored or disengaged) (Soleymani, Pantic, & Pun, 2012).

Because of the aforementioned issues, recent years have seen an increase in the use of field observation techniques for labeling instances of affect in real time (Baker et al., 2012; Botelho et al., 2017; DeFalco et al., 2018; Jiang et al., 2018). A benefit of field observations is that the annotations occur in real time within the learning environment, which makes the labeled data easier to obtain and can help the models maintain reliability and consistency, as the data was collected under real-world conditions (DeFalco et al., 2018). Additionally, the use of field observers ensures that the full context of the data collection and the surrounding environment is taken into account, as this context cannot always be reliably or fully captured by single modalities such as video or audio recordings. The establishment of inter-rater agreement prior to the annotation of leaner affect, such as within the Baker Rodrigo Ocumpaugh Monitoring Protocol (BROMP), ensures that a pre-defined level of agreement between multiple annotators is achieved during specific data collections.

While affective computing encompasses a wide variety of emotional states, educational contexts typically focus on a subset of academic emotions that are commonly present during

learning processes. This subset includes boredom, confusion, delight, engaged concentration, frustration, and surprise (Baker et al., 2010). Each of these emotions are typically either negatively or positively correlated with learning outcomes. For example, boredom often has a noticeable negative correlation with learning (Pekrun et al., 2010), and has even been associated with students "gaming the system" when interacting with various virtual learning environments (Baker et al., 2010). Engaged concentration is typically associated with increased learning gains (Pardos et al., 2014). However, confusion and frustration are more complex emotions that can be associated with both positive and negative learning outcomes.

The affective state of confusion has seen an increase in research focus due to its multi-dimensional associations with student learning (D'Mello et al., 2014; Lehman, D'Mello, & Graesser, 2012). In an exploration of the transitory patterns of various academic emotions, D'Mello and Graesser determine that a state of confusion is often the result of reaching an impasse within the learning process (2012). Following analysis of two studies where participants completed a tutoring session with an automated tutoring system for physics, it was discovered that confusion can lead to: 1) a state of engagement or flow should the impasse be resolved or overcome, or 2) a state of frustration should the impasse remain unresolved for an extended period of time. In related prior work, VanLehn et al. hypothesize that deep knowledge acquisition cannot be truly achieved without impasses and confusion (2003).

While frustration is a more negatively valenced emotion, Pardos et al. discovered that a positive correlation exists between frustration and student performance on an end-of-term mathematics exam following interactions with an intelligent tutoring system (2014). However, D'Mello and Graesser observed that a transition to a state of frustration could subsequently lead to a state of boredom, which had a negative correlation with student performance and coincided

with student disengagement with learning platforms (2012). Additionally, there are numerous studies that have established a negative correlation between frustration and learning gains (DeFalco et al., 2018; D'Mello & Graesser, 2011; Grafsgaard et al., 2014; Harley, Bouchet, & Azevedo, 2013). Other work was unable to determine an outstanding correlation between frustration and learning gain (Craig et al., 2004; Rodrigo et al., 2009). The complex relationships between these affective states and learning outcomes necessitates the accurate prediction of instances of these emotions, in order to enhance learning experiences and promote engagement with learning platforms through user-sensitive intervention mechanisms.

In theory, affect detection can typically be presented as a binary or multi-class prediction task, which proves a natural fit for machine learning approaches used for modeling behavioral patterns. These models are then subsequently used to predict certain quantifications of affect (Song et al., 2019). Affect detection tasks can typically be divided into two categories, sensor-free and sensor-based. Sensor-based affect detection relies on data captured by physical sensors to capture behavioral patterns and expression exhibited by a person in order to detect and classify instances of emotion. Examples of sensor-based modalities include facial expressions (Bosch, Chen, et al., 2015; Calvo & D'Mello, 2010; Grafsgaard et al., 2013a; Sawyer et al., 2017; Wei et al., 2016; Whitehill et al., 2014), posture (DeFalco et al., 2018; Grafsgaard et al., 2012; Henderson, Rowe, et al., 2019; Sanghvi et al., 2011; Wei et al., 2016), eye gaze (Rajendran et al., 2018; Thomas et al., 2018; Wu et al., 2019), and biometric modalities such as EDA (Calvo & D'Mello, 2010; Chen & Jin, 2015; Henderson, Emerson, et al., 2019; Kalimeri & Saitis, 2016) and EEG data (Calvo & D'Mello, 2010; he et al., 2015; Soleymani et al., 2016). Sensors have become more prevalent within affect detection frameworks due to their decreasing hardware costs, generalizability to various environments and domains, and relative ease of implementation. However, sensors are

also not practical or permissible within certain environments, and certain data channels such as video or audio capture raise privacy and ethical concerns. Additionally, sensors are prone to hardware failure, inconsistent behavior, mistracking or miscalibration, data storage constraints, and other issues that can lead to corrupted or missing data (Bosch, Chen, et al., 2015; Henderson, Emerson, et al., 2019; Jaques et al., 2017). Additionally, certain physical sensors (particularly biometric sensors such as EDA or EEG) require specialized equipment that must be positioned on a participant, such as a wristband or headset. The intrusiveness of these sensors can lead to scalability and deployment issues.

Alternatively, sensor-free affect detection systems bypass many of the issues prevalent in sensor-based approaches. Sensor-free data sources are usually integrated into the software of particular platforms the participant is engaging with, such as virtual learning environments (Grafsgaard et al., 2014) or game-based platforms (Min et al., 2017). The feature representations of sensor-free modalities are typically extracted from interaction trace log data from the software. An alternative to platform-specific trace data includes peripheral interaction data such as mouse movement or keystrokes (Sun et al., 2014). Text-based modalities extracted from sources such as dialogue-enabled virtual agents or discourse between multiple individuals have also become increasingly common within affective computing (Fung et al., 2016, Grafsgaard et al., 2014; Komatani & Okada, 2021; Ma et al., 2020). Sensor-free modalities are attractive due to their lack of intrusiveness, ability to capture data without causing distraction, reduced noise compared to sensor-based modalities, and independence from external hardware, thus reducing the risk of data loss due to inconsistent hardware behavior (Baker et al., 2012). Additionally, sensor-free systems are highly scalable, as there are no physical hardware constraints or associated hardware costs. However, sensor-free systems are not as easily generalizable, due to the need for platform-specific

software modifications to enable the logging of user interaction data. This aspect is also a factor in the feature engineering process, as many sensor-free implementations are reliant on domain-specific data representations used to induce the affect detection models. Additionally, the capture of user actions within a particular virtual environment are fairly infrequent compared to the sampling rate of physical sensors, resulting in a less data-rich modality.

2.2. Multimodal Affect Detection

Due to the increase in accessibility and affordability of physical hardware sensors, affect detection has seen an increased emphasis on multimodal modeling techniques. Multimodal approaches to affect detection have several notable benefits over unimodal approaches. The use of multiple modalities allows a model to consider multiple perspectives of a person's physical behavior and enables higher-fidelity models to be induced through multiple different data representations (D'Mello & Kory, 2015; Ghaleb et al., 2019; Worsley et al., 2015). Additionally, multimodal models can be trained to compensate for instances where a single modality suffers from hardware issues or other obstacles that result in missing data, so that affect detection can still occur (Bosch, Chen, et al., 2015). Recent work involving the use of a multimodal approach to affect detection varies widely with regards to 1) selecting which modalities to include, 2) combining the selected modalities, and 3) modeling the multimodal data.

DeFalco et al. compare the performance of a set of non-neural machine learning models when trained on posture and interaction trace data separately (2018). The models were trained to predict instances of particular states of affect that occur during student interactions with a game-based learning environment, with the ground-truth labels obtained via third-party field observers. It was observed that the models trained on the interaction logs outperformed the posture-only models by a significant margin for four of the five affective states evaluated (bored, confused,

engaged concentration, frustrated, surprised). However, this work is not considered multimodal as multiple modalities were not considered simultaneously. The work presented in this dissertation proposal extends this prior work by investigating multimodal modeling techniques on this dataset.

Grafsgaard et al. (2014) explore the effectiveness of multimodal models for the prediction of frustration, engagement, and learning gain using a mixture of sensor-based and sensor-free modalities. The ground-truth labels were collected using self-report surveys administered after students interacted with a virtual tutoring environment for teaching introductory programming. Textual interactions with the virtual tutor were captured as a dialog modality, while posture and gesture were captured with a Microsoft Kinect, which formed a sensor-based modality. Additionally, facial expressions were captured using a webcam mounted to each workstation. The facial expression features were combined with the posture and gesture modality to form a single "non-verbal" data channel. Finally, the user's actions within the virtual environment were logged and used as a task-oriented modality. Forward stepwise linear regression was used to construct the predictive models, and multiple modalities were combined using feature-level data fusion. The trimodal combination of all modalities outperformed bimodal and unimodal models across all three evaluations, demonstrating an additive benefit to combining multiple modalities.

Chen and Jin (2015) perform a similar multimodal analysis for modeling engagement in pairs of students using recorded audio, video, electrocardiogram (ECG) and EDA modalities, using a real-time annotation protocol with third-party field observers. The authors use Long Short-Term Memory (LSTM) recurrent neural networks to perform sequential modeling of the student behavior to estimate engagement valence and arousal levels. Additionally, the authors explore various multimodal data fusion approaches, including feature-level concatenation of features, a decision-level fusion with a linear regression model, and hybrid fusion by sharing latent data

representations between the LSTM's hidden layers based on the timestamp within a sequence. The results indicate that decision-level fusion was the optimal method for combining modalities when predicting engagement arousal, and feature-level fusion was the optimal method for predicting engagement valence.

Bosch et al. explore the multimodal interaction between a facial expression modality and an interaction-based modality captured from students engaged with a virtual learning environment for physics (Bosch, Chen, et al., 2015; Bosch et al., 2016). Instances of five academic emotions (boredom, confusion, delight, engagement, frustrated) were captured using third-party field observers. Both feature-level and decision-level fusion were considered for combining the modalities. Additionally, unimodal models were trained on both modalities to serve as baselines. For four of the five affective states, the facial expression-based models outperformed the interaction-only model, indicating that facial expressions are more predictive of student affect than interaction-only modalities. Additionally, the decision-level fusion outperformed the feature-level fusion models for three of the five affective states as well. The multimodal models displayed marginal improvement in predictive performance over the unimodal models, with the exception of confusion. However, the use of decision-level data fusion helped mitigate instances where a modality was invalid or missing (e.g., face mistracking) and demonstrated improved performance by using only intact data in such instances instead of removing the entire data sample.

Because of the rapid increase in literature on multimodal affect detection, there are many surveys of recent work in this field that have been conducted in order to analyze significant trends in models, modalities, and outcomes. D'Mello and Kory (2015) conducted a survey of 90 peer-reviewed multimodal affect detection systems and identify that 85% of the observed multimodal systems offered improved performance over their unimodal counterparts. The authors (2012) also

report a relatively even distribution between feature-level and decision-level multimodal data fusion, with 38.9% and 35.6%, respectively. In a separate study conducted across 30 unimodal and multimodal affect detection systems, the authors observe that the two most frequently used modalities are facial expressions (77%) and acoustic dialogue (77%). Additionally, it is noted that approximately 30% of the studies utilized body movement, posture, or gesture modalities. However, instances of eye gaze or biometric modalities such as ECG, EEG, or EDA data was noticeably rare. The authors note a shift towards the use of three or more modalities, while prior work primarily focused on the use of two modalities. However, at the time of publication, the survey notes that over 70% of the observed systems were bimodal, while almost all remaining systems were trimodal. The authors also noted that for many multimodal works, there was not a direct correlation between the number of modalities included and the predictive performance of the affect models. Other surveys of recent work in multimodal affect detection conclude that a vast majority (>90%) of multimodal systems focus on audio, visual, and textual modalities (Poria et al., 2017). While feature-level and decision-level fusion still remain the two predominant forms of multimodal data fusion, recent years have seen an increase in the use of more sophisticated methods, such as hybrid fusion or model-based fusion.

Multimodal systems often encounter two issues related to data capture: noisy, invalid, or missing data, and relatively small datasets. The use of certain modalities, particularly sensor-based modalities, introduces the risk of a sensor malfunctioning and corrupting an entire multimodal data sample. While a simple solution is to simply remove the entire data sample from the dataset, this results in reduced data availability and can adversely impact the affect model's predictive performance. Consequently, recent work has investigated various means of data imputation. Jaques et al. (2017) explore the use of denoising multimodal autoencoders as a means of imputing

missing values in 11 distinct modalities used to predict self-reported affect and stress levels. Yoon et al. (2018) explore the use of generative adversarial networks (GANs, Section 2.3.4) as a means to impute particular missing values, and preliminary work for this dissertation proposal explores the use of generative modeling techniques for the imputation of multimodal data used to enhance the performance of affect detection models (Henderson, Min, et al., 2020a).

Multimodal data is often challenging to capture, requiring all devices to be calibrated and operational, and for the observed user to be within the viewing frame of all sensors. As a result, data collections are often infrequent, and the amount of available multimodal data can be further reduced if issues occurred with one or more methods of capturing a particular modality, as mentioned in the previous paragraph. Additionally, a problem within affective computing is the difficulty with recording or annotating instances of affect, either through self-reporting or third-party field observations. For example, if a particular inter-rater agreement threshold is not met between multiple affect annotators, then the corresponding multimodal data is often declared invalid and is removed from the dataset. The resulting loss of data can have a noticeable impact on the predictive performance of multimodal affect models, particular deep learning-based models. In addition, certain affective states have been noted to be extremely infrequent, such as observed instances of surprise or anxiety. This leads to highly imbalanced data distributions for these particular affective states and can adversely impact performance as well. However, recent work has investigated various methods of generating additional synthetic data for the purpose of enhancing the training of various affect detection models. Zhu et al. (2018) explore the use of GANs as a means to generate synthetic images of facial expressions to enhance the performance of a convolutional neural network used to classify the images into 7 distinct groups of emotions. Similarly, Krokotsch and Böck (2019) use GANs to generate synthetic speech data to improve the

performance of a convolutional neural network trained to classify affect present in spoken dialogue samples. Qui and Zhao (2018) use auxiliary classifier GANs (Section 5.3.1.1) to generate denoised EEG data used to generate spatial representations for training a convolutional neural network to classify emotion. Auxiliary classifier GANs are investigated in our preliminary work as a means to generate synthetic multimodal training data to enhance the performance of our affect detection models.

2.3. Deep Learning

Recent years have seen a significant increase in the use of deep learning to solve computationally complex problems through their capability to model high-dimensional, non-convex patterns. Deep learning has seen many applications in a variety of fields, including natural language processing (Young et al., 2018), computer vision (Zhao et al., 2019), speech detection, (Deng, Hinton, & Kingsbury, 2013), and data imputation (Yoon et al., 2018). In the following sections, we present an overview of deep learning (Section 2.3.1) and introduce three principal variations of deep learning approaches, namely, deep feedforward neural networks (Section 2.3.2), autoencoder neural networks (Section 2.3.3), and GANs (Section 2.3.4). Finally, this section includes a brief overview of various computational tasks that have been investigated using deep neural networks (Section 2.3.5).

2.3.1. Overview of Deep Learning

Deep learning is a form of representation learning that relies on the use of multi-layered artificial neural networks (ANNs). ANNs are inspired by the information processing and distributed communication found in biological neurological systems. The first instance of a mathematical model based on a biological neuron was published by Pitts and McCulloch, which would

eventually form the theoretical foundation for ANNs and deep learning (1943). The first supervised learning algorithm for training deep, multi-layer perceptron networks was published in 1967 (Ivakhnenko & Lapa, 1967), which was followed by the introduction of backpropagation as an enhanced approach to ANN training (Werbos, 1974). Additional research in this field introduced the concept of momentum during backpropagation training, which further investigated the ANN's ability to extract internal, latent representations of data within the network's hidden layers (Rumelhart, Hinton, & Williams, 1986).

Following this period, research efforts related to deep learning in the late 1980's experienced a steep decline, due to existing limits in hardware processing capabilities and computational issues within neural network architectures such as vanishing and exploding gradients (Hochreiter, 1998). However, deep learning experienced a renaissance in the late 1990s through the present day, due to a number of factors: 1) increased computing capabilities through graphical processing units (GPUs) and advances in parallel computing, 2) increasing amounts of readily available labeled and unlabeled datasets, 3) unsupervised pre-training techniques such as deep belief networks (Hinton, Osindero, & Teh, 2006), 4) advances in software infrastructure capabilities, and 5) the introduction of model training techniques including the ADAM optimization method (D. Kingma & Ba, 2017) and regularization approaches such as dropout (Srivastava et al., 2014) or artificial noise injection (Poole, Sohl-Dickstein, & Ganguli, 2014; Vincent et al., 2010). This period of expansive growth has produced a number of advanced deep learning modeling approaches including stacked denoising autoencoders (Vincent et al., 2010), variational autoencoders (Kingma & Welling, 2014), convolutional neural networks (LeCun et al., 1989), recurrent neural network (RNN) models such as LSTM networks (Hochreiter & Schmidhuber, 1997), and generative adversarial networks (Goodfellow et al., 2014).

2.3.2. Deep Feedforward Neural Networks

The foundational neural network architecture is the feedforward neural network (FFNN). The FFNN consists of an input layer, one or more hidden layers, and an output layer. Each layer is comprised of one or more neurons or "perceptrons", which results in FFNNs often being referred to as "multi-layer perceptrons". Each neuron is connected to the neurons in the following layer by a single weighted connection, an architecture known as "fully connected layers". Each neuron is typically activated by a non-linear activation function (e.g., *sigmoidal* or *hyperbolic tangential* function), which is the construct that enables the network to model complex, non-linear functions. The input layer is encoded based on the dimensions of the feature vectors used to train the network.

Currently, the most widely used method of training FFNNs is backpropagation, which is the process of quantifying the performance (or "error") of the network through a loss function (e.g., cross-entropy or mean squared error), and adjusting the weights of the network by propagating the quantified error from the output layer to the input layer. Each individual weight in the network is adjusted according to the gradient descent method, which computes the derivative of the error with respect to the weights.

2.3.3. Autoencoder Neural Networks

Another type of deep learning model is the *autoencoder*, which is an ANN that is trained to reconstruct its input as the output. During this process, the autoencoder learns a "hidden" (i.e., "latent") representation of the input in order to reconstruct the output. The dimensionality of the latent representation of the input is usually smaller than the original input, which has led the autoencoder to be a common approach to dimensionality reduction as part of a broader data modeling pipeline. The autoencoder typically consists of two components, the *encoder* and the *decoder*. The encoder takes an input x and uses a learned mapping function m to produce a latent

space representation $\boldsymbol{h} = m(x)$. The decoder attempts to reconstruct x from h using a decoding function r so that $\hat{\boldsymbol{x}} = r(h)$. Autoencoders typically consist of fully connected layers within the encoder and decoder components, although recent work has also explored the use of convolutional layers (Chen et al., 2017) and sequential models such as LSTMs (Tu et al., 2018). Training of the autoencoder is typically implemented using backpropagation in a manner similar to that of FFNNs, with the objective of minimizing a reconstruction loss function $\boldsymbol{L} = (x, \hat{x})$. A visualization of a standard autoencoder neural network is shown in Figure 2.1.

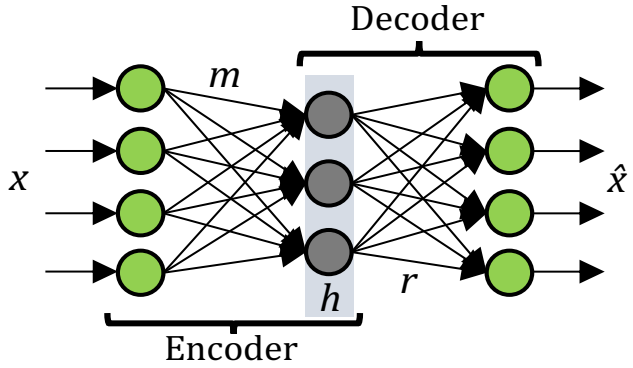

Figure 2.1. Visualization of autoencoder neural networks.

In recent years, there have been several variations of autoencoders that have been introduced. Examples include denoising autoencoders, which seek to enhance the performance of the model through the use of artificial noise injection into the input. This process enables the autoencoder to be more robust against noise by training the model to reconstruct the original, uncorrupted data (Vincent et al., 2010). Another example is a variational autoencoder (VAE), which is a generative model whose latent representation of the input data is trained to model parameters of a pre-determined probability distribution. This is accomplished through a reconstruction loss that consists of the reconstruction loss in addition to the Kullback-Leibler divergence, which quantifies the distance between the modeled probability distribution and a given

target distribution and allows for the generation of augmented data based on the modeled probability distributions (Kingma & Welling, 2014).

2.3.4. Generative Adversarial Networks

GANs consist of two neural network components, a *generator* and a *discriminator*, that "compete" against one another other in an adversarial fashion (Goodfellow et al., 2014). The primary objective of a GAN is to learn to accurately generate synthetic data that closely resembles a provided data distribution by training the two networks within a framework similar to a zero-sum game. Using a random noise vector as input, the generator attempts to produce synthetic data that deceives the discriminator, which subsequently attempts to determine whether its input is synthetic (i.e., "fake" data) or sampled from the provided data distribution (i.e., "real" data). The loss of the discriminator is iteratively backpropagated through both components of the GAN, with the goal of teaching the generator to produce increasingly realistic augmented data, while the discriminator also learns to accurately distinguish between the real and fake data samples. GAN training convergence is theoretically achieved when the model achieves a Nash equilibrium (Arora et al., 2017). While GANs are often implemented using convolutional neural networks, these networks can also be implemented using standard FFNN models for the generator and discriminator as well. A visualization of the training process for a GAN is shown in Figure 2.2.

2.3.5. Recent Applications of Deep Learning

Deep learning techniques have produced state-of-the-art results in a number of diverse research efforts and tasks. Common neural network architectures, including convolutional neural networks, RNNs, and FFNNs, have demonstrated optimal performance in fields such as automated speech

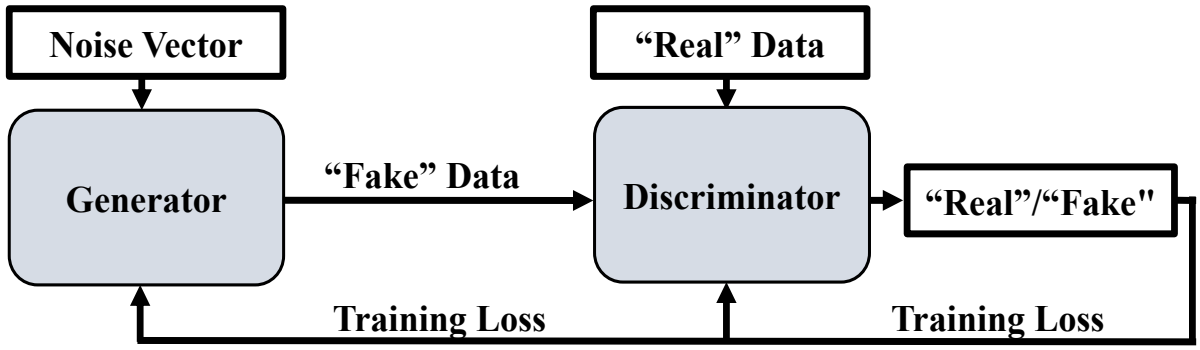

Figure 2.2. Visualization of GAN training process.

detection (Deng & Platt, 2014), image detection (Zhong et al., 2018), natural language processing (Young et al., 2018), recommender systems (Elkahky, Song, & He, 2015), domain adaptation (Tzeng et al., 2017), machine translation (Sennrich, Haddow, & Birch, 2016), and reinforcement learning (Arulkumaran et al., 2017).

Autoencoders have been shown to achieve optimal performance in a number of tasks as well. A common application of autoencoders is dimensionality reduction, as the autoencoders' ability to learn non-linear latent representations of high dimensionality shows these networks to outperform traditional methods such as Principal Component Analysis (PCA) (Hinton & Salakhutdinov, 2006). However, more recent research efforts have seen autoencoders applied to an expanding number of domains. Jaques et al. (2017) demonstrate the effectiveness of a denoising autoencoder as a means to perform multimodal data imputation, while Zhou and Paffenroth (2017) similarly adapt denoising autoencoders as an approach for unsupervised anomaly detection through various regularization techniques. Tu et al. (2017) apply a supervised autoencoder to an end-to-end neural machine translation framework to enhance the reconstruction of a sentence from a source language to a target language. The introduction of variational autoencoders enabled these models to be used within generative modeling tasks, including improving speech detection through

data augmentation (Hsu, Zhang, & Glass, 2017), semi-supervised text classification (Xu et al., 2017), and speech generation (Akuzawa, Iwasawa, & Matsuo, 2019).

Similarly to variational autoencoders, GANs are primarily used for generative modeling, such as facial image generation (Karras et al., 2018), creation of game levels (Park et al., 2019), and data augmentation (Goodfellow et al., 2014). However, this architecture has also been applied to additional tasks such as image-to-image translation (Isola et al., 2017) and text-to-image translation (Zhang et al., 2017). Liu and Tuzel (2016) explored the effectiveness of coupled GAN models as an approach to adversarial domain adaptation, while Yoon et al. (2018) utilize a GAN-based framework as an alternative method for missing data imputation through the use of hint generation within the adversarial framework.

CHAPTER 3

DEEP LEARNING-BASED MULTIMODAL AFFECT DETECTION FOR ADAPTIVE LEARNING ENVIRONMENTS

This dissertation details a multimodal affective detection framework that ultimately predicts the current affective state of students based on various input channels. The multimodal data is captured from students engaged with digital learning environments through the use of physical sensors and software-based mechanisms. The framework is intended to utilize a variety of data channels including interaction-based (e.g., gameplay trace data, keystroke data), behavior-based (e.g., posture, gesture, facial expression), and physiology-based (e.g., EEG, EDA) modalities. Following the capture of the raw multimodal data, feature engineering is performed on each individual modality and the resulting features are used to induce affect detection models. The features from each modality are combined using different model-agnostic multimodal data fusion techniques, namely decision-level and feature-level data fusion. Following this process, the predictions generated by the trained student affect models can be employed to enable user-sensitive mechanisms integrated within adaptive learning environments. Examples of these mechanisms can include dynamic hint generation, personalized feedback, and in-game challenges tailored to individual students' tendencies, skills, and performances. The desired outcome of enabling such user-adaptive learning environments is to maintain user engagement, improve emotion regulation, and promote knowledge acquisition.

This dissertation addresses three core components: multimodal affective modeling, data sparsity, and generalizability (Figure 3.1). The multimodal affective modeling component includes aspects of the framework's predictive capacities such as the systematic combination of multiple

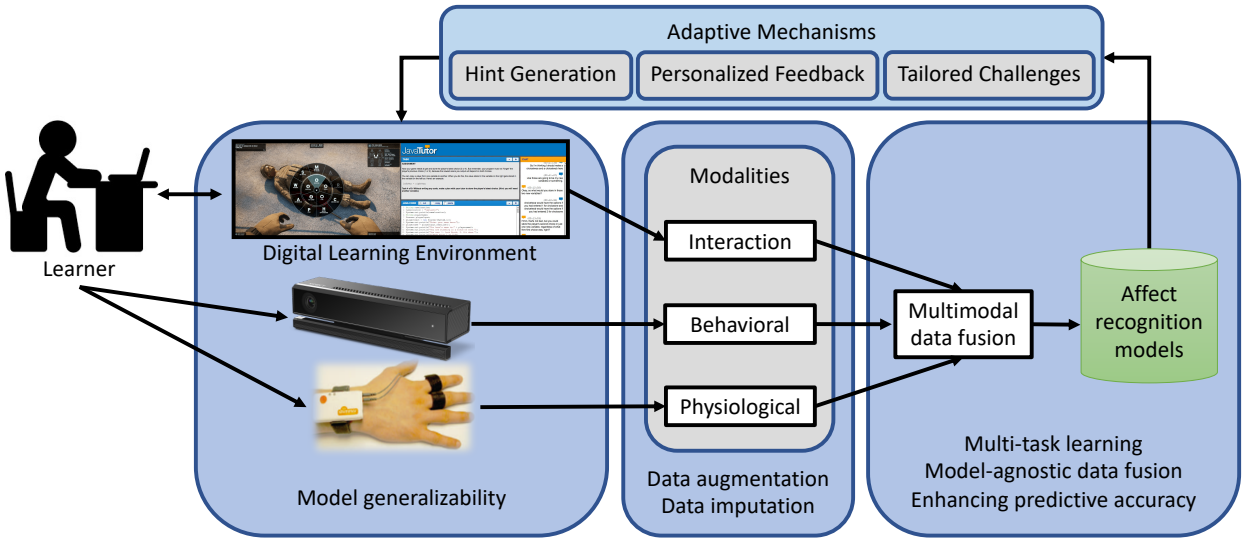

Figure 3.1. Proposed multimodal affect detection framework.

modalities and the evaluation of various deep learning-based approaches that improve the framework's predictive accuracy. The data sparsity component is targeted towards addressing issues that arise with limited or missing data from one or more modalities through data augmentation and data imputation, respectively. Similar to the aforementioned component, this portion of the dissertation also evaluates deep learning-based approaches to generating synthetic training data or imputing missing values within the multimodal dataset, with the overarching goal of improving the affect detection models' predictive performance by expanding the available training dataset. Finally, the generalizability component of the dissertation provides additional evidence that the approaches and methods described in this work can be effectively deployed across different learning environments, modalities, student populations, and instructional platforms through empirical evaluations on multiple multimodal datasets.

The work reported in this dissertation includes the evaluation of deep learning-based approaches to data augmentation, data imputation, and affect detection using interaction-based (gameplay and interaction trace data, dialogue utterances), behavior-based (posture, facial

expression), and physiology-based (EDA) modalities. Additionally, variations of model-agnostic multimodal data fusion were evaluated to determine their impact on the affect detection models' predictive performance. The preliminary work was performed using multimodal data captured from students engaged with a game-based learning environment for providing training in emergency medical care, as described in Chapter 4. The results provide support for the effectiveness of deep learning-based techniques for data augmentation and imputation compared to a series of non-neural baseline models. Additionally, deep learning models were shown to be effective as affect detection models for a variety of affective states as well. It is noted within the results that the combination of the interaction-based and behavior-based modalities yielded stronger results than the interaction-based and physiology-based modalities. Additional research described in Chapter 11 includes an investigation of more complex approaches within the affective modeling component, namely multi-task learning. This work addresses the generalizability of the separate elements of the framework by evaluating the same deep learning-based approaches. For each component, the approaches presented in Chapters 7, 8, 9, 10, and 11 are also evaluated across another data corpus that consists of two sensor-free modalities (dialogue utterances, interaction trace data) and one sensor-based modality (facial expression) to demonstrate the generalizability of the different modeling techniques. This separate data corpus is comprised of modalities captured from students engaged with a computer-mediated web-based integrated development environment to teach introductory programming, and is further described in Chapter 4. These generalizability evaluations extend to the data augmentation and data imputation across various multimodal data, and also include the evaluation of different combinations of multimodal data fusion techniques and affect detection modeling approaches. The final results of the dissertation lay the foundation for the integration of different user-sensitive mechanisms within adaptive learning environments,

although the development and evaluation of such mechanisms are not included within the scope of this work.

CHAPTER 4

MULTIMODAL AFFECT DETECTION DATA CORPORA

This chapter presents the testbed platforms and multimodal data corpora previously captured from student interactions with different digital learning environments. The first dataset (USMA dataset) was captured from students engaged with a serious game used for training cadets in administering tactical combat casualty care, *TC3Sim*. The students' posture as well as their in-game actions and characterizations were captured during their interactions with *TC3Sim*, with the resulting posture and interaction trace data being used to engineering various feature representations to induce the various affect detection models. Additionally, electrodermal activity of each student was captured as well. The second dataset (*JavaTutor* dataset) was captured from undergraduate students interacting with tutors through a computer-mediated development environment designed to teach introductory programming, *JavaTutor*. The students' activity and keystroke data were captured, in addition to the text-based chat dialogue from both the tutor and the student. The *JavaTutor* data also consists of a single sensor-based modality extracted from the facial expression features of each student. By capturing multimodal representations of a student's physical behavior in addition to their interaction with the different learning environments, we aim to accurately model complex relationships between multimodal student data and occurrences of different affective states.

Section 4.1, particularly Sections 4.1.1-4.1.4, provides a brief overview of the USMA dataset, including the implementation of the *TC3Sim* testbed, the prior data collection procedures, and the various interaction- and sensor-based modalities previously captured while the students interacted with *TC3Sim*. Additional detail is provided about the annotation process for instances of student affect, as well as the resulting distribution of the aforementioned affective states captured during the study. Section 4.1.5 provides additional insight into the feature engineering

process following the initial data capture, including the feature representations used for the posture, EDA, and interaction modalities. This section also provides an overview of the generation of additional features based on the velocity of certain postural vertices to provide additional temporal context to the intensity of shifts in a student's posture throughout their gameplay session. Similarly, Section 4.2 provides an overview of the *JavaTutor* data corpus, including additional information on the *JavaTutor* programming environment, the student population that participated in the study, multimodal data capture, and the method of self-reporting affect information following each learning session. Section 4.2.5 provides additional detail related to the feature engineering process for the interaction-, dialogue-, and facial expression-based modalities.

4.1 USMA Data Corpus

The data collection took place at The United States Military Academy (USMA) in 2013. During this study, military cadets completed a series of training materials on emergency medical care, including interacting with a serious game-based learning environment (*TC3Sim*). While the students interacted with this learning platform, instances of pre-determined affective states were annotated by two field observers. In addition, the students' posture, electrodermal activity, and in-game actions were recorded through the use of various physical sensors and a specialized software framework used for developing and deploying adaptive training software. The resulting dataset consisted of multimodal input data channels that were used to induce various affect models trained to accurately predict the annotated affect instances recorded by the field observers.

4.1.1 *TC3Sim* Backstory and Gameplay

Developed by Engineering and Computer Simulations (ECS), *TC3Sim* is widely used by the U.S. Army to train soldiers in the essential procedures required in emergency medical care. In *TC3Sim*,

trainees complete a series of 3D simulated combat missions alongside a group of non-player characters (NPCs). The story-driven training scenarios include a series of simulated combat events that lead to the eventual injury of one or more teammates. Players were free to navigate the open-world environment in *TC3Sim* as they pleased; however, the events and scenarios in *TC3Sim* were linearly structured (DeFalco et al., 2018). The first scenario was a tutorial that introduced the controls and game mechanics of *TC3Sim*. The next scenario was a relatively simple task involving a leg amputation and the application of a tourniquet. The third scenario centered on a squad that goes on patrol but is ambushed, leading to an amputation task and a bullet wound task. The final scenario involved a victim with multiple severe hemorrhages requiring immediate medical attention. The victim expired regardless of the actions undertaken by the user.

4.1.2 Study Description

The data collection took place in September 2013 using a group of 119 first-year cadets (83% male, 17% female, aged 18-22) at the United States Military Academy in West Point, New York. The study took place within a classroom with each cadet assigned to an individual workstation consisting of a laptop and several physical sensors to capture the physical behavior of the cadets as they interacted with *TC3Sim*. Ten separate workstations were setup during this study, with each cadet assigned to their own individual workstation. Prior to using *TC3Sim*, each participant completed a brief content pre-test and viewed a PowerPoint presentation explaining the medical tactics, skills, and procedures that would be practiced in the game (e.g., tactical field care, hemorrhage control). Following the PowerPoint presentation, cadets interacted with *TC3Sim* by completing the four training scenarios mentioned in Section 4.1.1. The training materials (pre-survey, PowerPoint, sensor calibration, post-survey, etc.) were deployed using the Generalized

Intelligent Framework for Tutoring (GIFT), a service-oriented framework for developing and deploying user-adaptive training platforms (Sottilare et al., 2018).

Each cadet's posture was captured using a front-facing Microsoft Kinect for Windows v1.0 sensor. The Kinect was intended to capture the physical posture and gestures of a student throughout the gameplay session and was mounted to a tripod positioned in front of each workstation. Additional biometric data was captured using an Affectiva Q-Sensor, which is an arm bracelet that was worn by each individual. The Q-Sensor captured the EDA data (i.e., skin conductance), skin temperature, and the (x, y, z) coordinates of the sensor according to an integrated accelerometer. For the purposes of this study, only the EDA data was analyzed from the Q-sensor. Finally, each student's in-game activity was captured using GIFT in the form of timestamped interaction trace data logs. This includes particular gameplay actions taken by the user, such as performing a check of a patient's vitals, requesting an evaluation, or performing a blood sweep. Additional data captured in the interaction data logs include changes in the condition of NPC characters encountered during the game, such as changes in blood volume and heart rate. An example of *TC3Sim* gameplay is shown in Figure 4.1 below.

4.1.3 BROMP Protocol

Timestamped labels of student affect were collected using the Baker Rodrigo Ocumpaugh Monitoring Protocol (BROMP) (Ocumpaugh et al., 2015). BROMP is a coding procedure designed to produce quantitative field observations (QFOs) of student affect and behavior. During the study, two observers walked around the perimeter of the classroom and discreetly annotated observed instances of particular affective states using a hand-held Android device and specialized field-observation software called HART. Annotations of student affect occurred in 20-second intervals

Figure 4.1. *TC3Sim* game-based learning environment.

and were intended to be captured as discreetly as possible to minimize the influence of the observers' presence and disruption of the students' gameplay. Prior to this study, the two observers established an inter-rater agreement exceeding 0.6 in terms of Cohen's Kappa (Cohen, 1960). For this work, six distinct affective states were captured: *bored, confused, engaged concentration, frustrated, surprised,* and *anxious.*

4.1.4 Corpus

Between the two BROMP observers, 3,066 observations were recorded. Any observations indicating disagreement between the observers were removed from the dataset, in addition to observations that occurred outside of the actual *TC3Sim* gameplay. This resulted in a final dataset

consisting of 755 labeled affective states. 435 of the BROMP observations were labeled as *engaged concentration* (M = .576, SD = .239), 174 as *confused* (M = .231, SD = .185), 73 as *bored* (M = .097, SD = .161), 32 as *frustrated* (M = .042, SD = .182), 29 as *surprised* (M = .038, SD = .045) and 12 as *anxious* (M = .016, SD = .089). Due to the low number of instances of *anxious*, we do not include this affective state in any of our analyses.

4.1.5 Feature Engineering

Following the capture of the raw interaction- and sensor-based data, static representations of the students' behavior leading up to each BROMP observation were generated for each modality. Because each BROMP observation occurred in 20-second intervals, each feature represented a maximum of 20 seconds of activity. By using statistical summative representations of the data extracted from each data channel, the modalities are aligned, and a multimodal feature vector is able to encompass multimodal capture of student behavior across a single time interval. Additionally, we generate a new set of temporally based features to provide additional context on the intensity of students' postural shifts over time; this process is described in further detail in Section 4.1.5.4.

4.1.5.1 Posture-Based Features

The Kinect sensor captured 3D coordinate data for 91 distinct vertices. Based on prior work related to posture-based affect detection, three vertices were selected from which 73 posture-based features were distilled, *top_skull, center_shoulder*, and *head* (Grafsgaard et al., 2012). Each of the features was computed based upon the students' posture and movement prior to the given BROMP observation. Each vertex produced 18 statistical features, including features such as most recent observed distance, most recent Z-coordinate value, minimum and maximum observed distance,

median observed distance, and variance in the observed distances. Distance was defined as the Euclidean distance between the vertex and the Kinect sensor. Additionally, summative features were calculated for each vertex using the minimum, maximum, median, and variance in distance observed across the preceding 5, 10, and 20 seconds prior to each BROMP observation. In addition to these 54 features, several features were generated to provide the total change in position and distance from the Kinect sensor over the prior 3 and 20 seconds. Finally, features were extracted to indicate whether the student was leaning forward, backward, or sitting upright based upon the median distance of the *head* vertex for each individual workstation and the current position of the *head* vertex. These three features were calculated across time windows of 5, 10, and 20 seconds, as well as the entire gameplay session up to the current BROMP observation.

4.1.5.2 Interaction-Based Features

The interaction log-based features were extracted from students' interaction (i.e., gameplay) trace logs from *TC3Sim*. These features represented the students' actions within the game as well as information about the virtual patients that received treatment during each training scenario. Features representing the states of the virtual patients included changes in systolic blood pressure and heart rate, exposed wounds, and lung volume. Features were also generated based on students' gameplay actions such as checking a patient's vital signs or requesting a medical evacuation. Each of the features was calculated cumulatively over the preceding 20 seconds prior to a BROMP observation and reported in terms of the sum or current count of a certain action. Additionally, measures such as the virtual patient's blood pressure were reported using the standard deviation or average. This process produced 39 distinct interaction log-based features.

4.1.5.3 EDA-Based Features

The Q-Sensor returned data consisting of a timestamp and an EDA reading. In a similar fashion to the Kinect posture data, summary statistics were calculated for the EDA modality including attributes such as minimum EDA, maximum EDA, variance of EDA, and median EDA. These statistics were also calculated across time windows of 5, 10, and 20 seconds prior to the BROMP observation. Additionally, the total change in the EDA readings across time windows of 3 and 20 seconds was calculated, resulting in 18 EDA features.

4.1.5.4 Temporal-Based Features

Temporal posture features were computed from the spatial posture feature vectors using the distance between two posture coordinates (Sanghvi et al., 2011). Using the *head* vertex, for each set of (x, y, z) posture coordinates, the coordinate deltas across two consecutive sensor readings were calculated. The deltas were used to calculate *velocity features* averaged across time windows of 3, 5, 10, and 20 seconds. For each posture coordinate, the mean, median, max, and variance of the average corresponding velocity were calculated. This process provided an additional 48 temporally related posture features. Due to the large number of additional features calculated per vertex, velocity information was not calculated for *center_shoulder* and *top_skull*.

4.2 *JavaTutor* Data Corpus

The *JavaTutor* data corpus was collected during the 2011-2012 academic year and consists of multimodal data captured from undergraduate students that interact with the *JavaTutor* platform, a web-based learning environment designed to teach introductory programming concepts. Students are presented with a series of exercises based in the Java programming language and interact with an expert tutor in real-time through a text-based chat interface. During the study, the students'

facial expressions were captured through a front-facing webcam. Additionally, the *JavaTutor* platform recorded all keystrokes and actions the student performed within the learning environment, as well as all dialogue that occurred between the student and the tutor. After each learning session, students completed post surveys designed to capture retrospective self-reports of *engagement* and *frustration* that occurred during the session, which serve as the two target variables for the multimodal affect detection models induced using the previously mentioned modalities.

4.2.1 *JavaTutor* Programming Environment

JavaTutor is a web-based programming environment designed to facilitate human-human interactions between a student and a tutor that form the basis for adaptive scaffolding for cognitive and affective support (Wiggins et al., 2017). The computation concepts introduced by the platform begin with introductory concepts such as programming statements and variables and progressive advance to control flow constructs such as nested if-statements and for loops. Each student is presented with a programming exercise that instructs the student to use the integrated development environment to write functional Java code to complete a given task. The tasks presented to each student were centered on the design and implementation of a text-based adventure game (Wiggins et al., 2017). The *JavaTutor* interface consists of four components: the description of the current programming exercise, the student's current Java code, the output of the most recent compilation or run attempt, and the cumulative chat messages sent between the student and the tutor (Mitchell et al., 2013). The *JavaTutor* environment allows students to write and compile Java code and run the program following a successful compilation. The tutor is able to observe all actions and code performed by the student during the learning session, as well as advancing the student to the next task. A screenshot of the *JavaTutor* interface is shown in Figure 4.2.

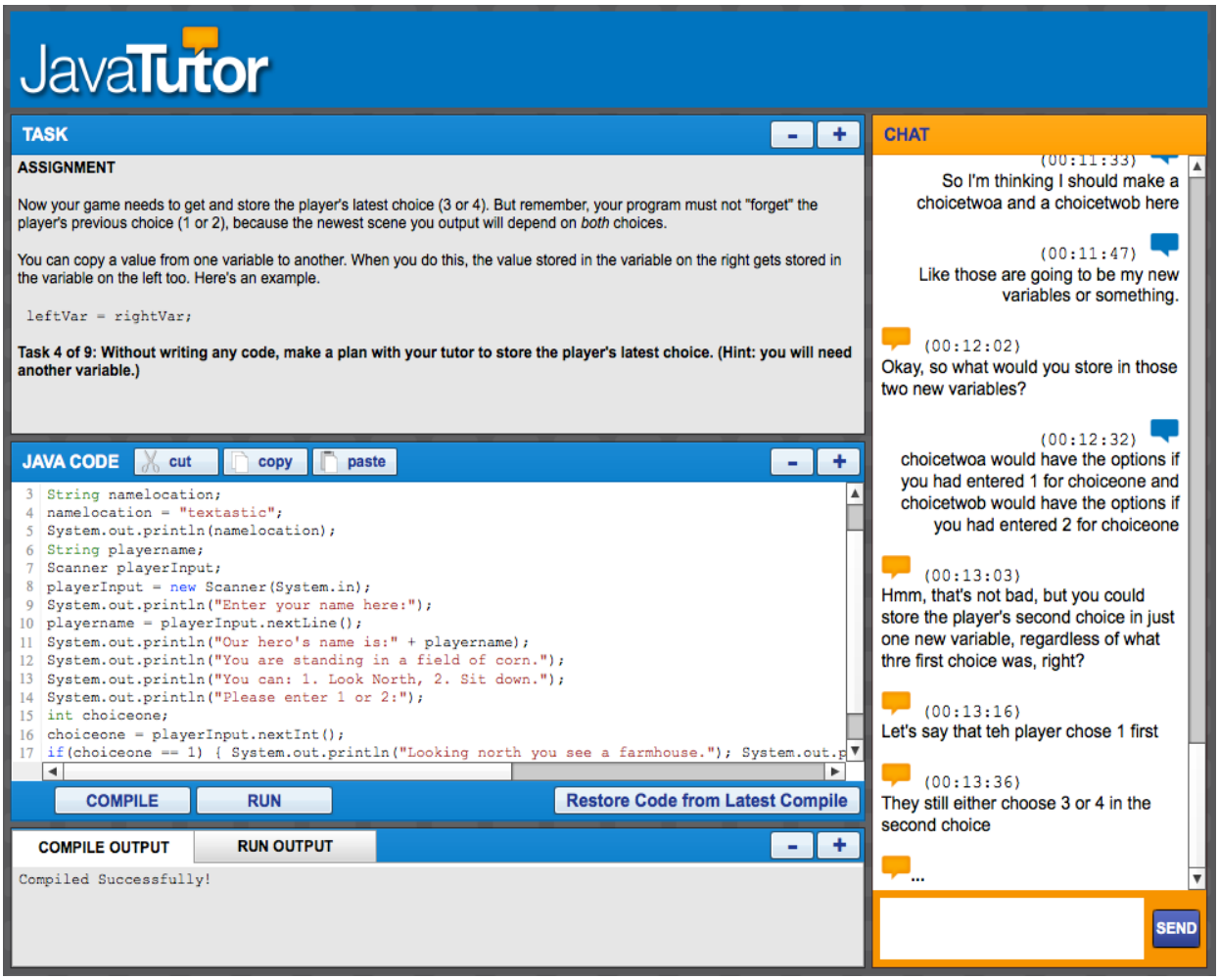

Figure 4.2. *JavaTutor* web-based programming interface.

4.2.2 Study Description

The data collection took place during a four-week period within the 2011-2012 academic year. 67 students (36% female, 64% male, average age = 18.5 years) from an introductory engineering course participated in the study, in addition to 5 human tutors that were primary graduate students with prior experience with providing instructional support in introductory computer science courses. Participants were pre-screened to eliminate students with significant prior experience with programming, and additional prior content knowledge was quantified using a pre-survey

administered prior to each learning session. Each student completed a series of six learning sessions that each lasted approximately 40 minutes.

A front-facing webcam was used to capture video recordings of each student for the duration of each learning session. In a post-processing step, facial action units (AUs) were extracted from the raw video files using the Computer Expression Recognition Toolbox (CERT) (Littlewort et al., 2011). The facial AUs used in this study are further detailed in the Facial Action Coding System (FACS) (Ekman, Friesen, & Hager, 2002). Finally, each student's activity during each learning session was stored in timestamped trace logs. This includes compilation successes or failures, as well as the code in the development environment recorded at each keystroke. Additionally, the messages sent between the student and the tutor are recorded and timestamped.

4.2.3 Affect Self-Reporting

Following the completion of each learning session, a student was administered two post-test surveys that included questions to determine the levels of *engagement* and *frustration* that the student experienced. Each student self-reported their *engagement* and *frustration* using the User Engagement Survey (UES) (O'Brien & Toms, 2010) and the NASA-TLX workload survey (Hart & Staveland, 1988), respectively. The UES survey consisted of 14 questions that were based on a 5-point Likert scale, while the NASA-TLX survey contained a single frustration-based question that was reported on a 0-100 scale. The questions that were used in each survey can be found in prior work based on the same *JavaTutor* corpus that investigated student *engagement* and *frustration* performed by Grafsgaard et al. (2014).

4.2.4 Corpus

After removing students that were either missing facial expression data, interaction data, or post-test survey results, 66 students remained. 347 individual learning sessions were completed between these students, in addition to completed *engagement* and *frustration* survey items. As a result, the dataset consists of 347 distinct affect labels for *engagement* and *frustration*, respectively. Histograms displaying the distributions of *engagement* (M=3.86, SD=0.59) and *frustration* (M=15.6, SD=20.9) are shown in Figures 4.3 and 4.4, respectively.

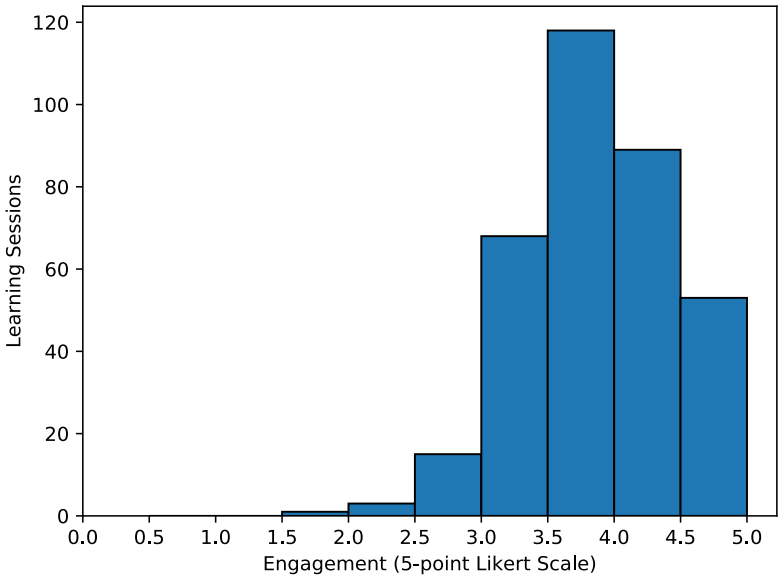

Figure 4.3. Histogram of student *engagement* in JavaTutor corpus.

4.2.5 Feature Engineering

Following the capture of the raw interaction and dialogue trace logs and the extraction of the facial expression features from the video recordings, features were generated from each modality that were used to train the multimodal affect detection models. Because only 347 data points exist for each affective state, student-level features are too sparse to sufficiently train a deep learning affect model. To mitigate this problem, we generate summative features across fixed time intervals,

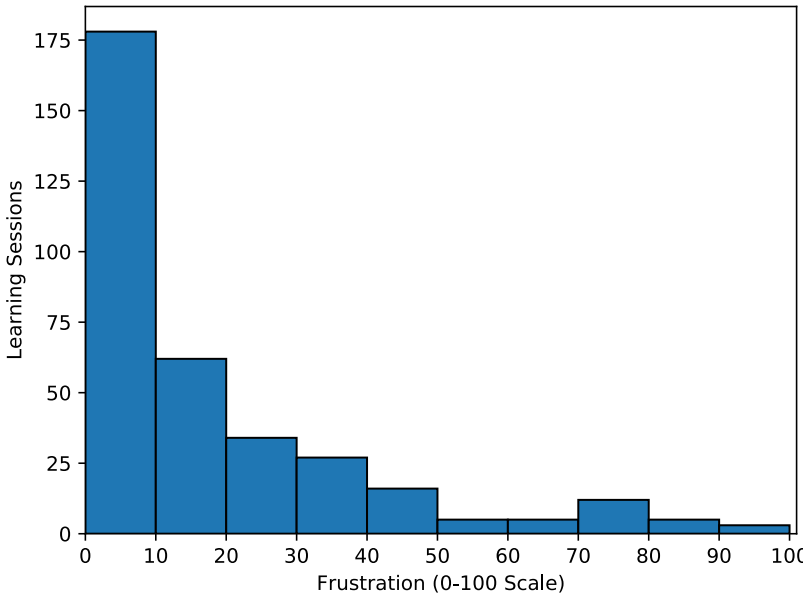

Figure 4.4. Histogram of student *frustration* in JavaTutor corpus.

where each feature vector contains either summed or averaged values from t=0 to t=n where n is the number of elapsed minutes since the start of the learning session, using a time interval of 10 minutes. We select this time interval so that each feature vector captures a sufficient number of interactions from all modalities and because individual learning sessions often lasted close to an hour. Summative features within each modality are normalized by the total elapsed time to avoid data leakage due to monotonically increasing features over time. As a result, the dataset consists of 1,324 different feature vectors. To generate a single prediction for a learning session, all summative feature vectors for a session are forward propagated through a trained model, and then the outputs are averaged to obtain a single prediction; therefore, the predictive performance of the models using the *JavaTutor* corpus is calculated using the student-level predictions.

4.2.5.1 Interaction-Based Features

Seven interaction-based features were generated from the trace data logs captured by the *JavaTutor* platform: 1) the number of compile errors, 2) the number of successful compilations, 3) the number

of completed tasks, 4) the number of successful program runs, 5) the number of student messages sent via chat, 6) the number of tutor messages sent via chat, and 7) the total amount of time the student spends interacting with the development component of the *JavaTutor* interface.

4.2.5.2 Facial Expression-Based Features

Using the processed facial AU data from CERT, we calculated the duration that each AU was exhibited throughout students' interactions with *JavaTutor*. We first standardized each student's observed AU intensity values and then calculated the duration of each AU during time intervals where consecutive AU intensity values were at least one standard deviation greater than the mean of that particular student-specific AU feature. To protect against data leakage when calculating the standard deviation, we only take into account the values from the beginning of the learning session up to the current time interval. This also provides a closer resemblance to a real-time modeling scenario where a student's entire learning trajectory may not be available when calculating the AU durations. This filtering process ensured that only spikes relative to the specific student's AU values contributed towards the calculation of the total duration. To further filter the AU durations, we only recorded the duration if the AU was present for longer than 0.5 consecutive seconds. This avoided possible micro-expressions that could add noise to the overall data channel (Sawyer et al., 2017). We performed this filtering process for all 19 AUs extracted by CERT. As a result, we extracted and distilled 19 facial expression-related features in total.

4.2.5.3 Dialogue-Based Features

Preprocessing steps for the raw dialogue text include removing extra whitespace and converting text to lowercase. Following this, we ran an algorithm to correct any perceived misspellings in the data, which uses the Levenshtein Distance algorithm to find any permutations of a word that are

within 2 edits of the original word. All of these permutations are compared to known words in a word frequency list, and it is assumed that more frequently-occurring words are most likely to be the correct spellings. Finally, common stopwords are removed from each sentence, as these are extremely frequent and do not hold predictive value for the machine learning models. Word embeddings were generated from a pre-trained BERT model for each word in a sentence, and then the embeddings were averaged across each utterance to obtain a single vector representation. Because the dimensionality of the word embeddings is often high (768 for BERT embeddings, for example), we use Principal Component Analysis on the sentence-level embeddings to reduce the dialogue representations to 20 components. Am empirical investigation into the selection of BERT as the method of generating word embeddings is detailed in Chapter 5, as well as additional analysis into the impact of different word embedding techniques on the predictive performance of multimodal affect models.

CHAPTER 5

DIALOGUE-ENHANCED MULTIMODAL MODELING OF STUDENT AFFECT

Dialogue has been shown to be highly indicative of student affect (Grafsgaard et al., 2014) and performance (Geden et al., 2021), and behavioral cues present in dialogue utterances such as sentiment and on-task/off-task discourse (Carpenter et al., 2020) can help inform such affect models. However, optimal machine learning methods for each modeling task are often open-ended issues, and questions remain regarding the optimal method of representing student dialogue, particularly in instances where manual annotation of dialogue utterances is impractical or prohibitive (Geden et al,. 2021). The *JavaTutor* corpus consists of 45,904 utterances between students and tutors across all learning sessions, which renders manual annotation impractical for the purposes of this work. As a result, we investigate computational approaches to word embedding representations for the affect models and draw conclusions on the predictive value of the dialogue modality through comparisons with a unimodal approach using the interaction-based modality only. Section 5.1 investigates several different neural and non-neural word embedding techniques frequently found in natural language processing literature, evaluates their impact on the predictive performance of multimodal affect models, and investigates the effectiveness of separating the word embeddings by source (student vs. tutor). Section 5.2 describes a post-hoc analysis that investigates the impact of different dialogue patterns on the multimodal affect models' performance, particularly regarding misclassifications of student *engagement* and *frustration* during particular time intervals within particular learning trajectories. This section also briefly details some of the limitations of this study, pertaining to aspects such as the data, feature representations, and natural language processing for affect detection. Finally, this section also

provides an overview of some of the prerequisites for implementing the dialogue-based multimodal framework in other learning environments outside of the *JavaTutor* platform.

5.1 Word Embedding Evaluations

The first aspect of this work investigates different predictive modeling approaches using the multimodal dataset, with particular emphasis on the dialogue modality. A number of neural and non-neural machine learning models are evaluated for their predictive performance for the two affective states, such as random forest, Naïve Bayes, logistic regression, and feedforward neural networks.

During this investigation, particular attention is given to the dialogue modality. Because this modality contains over 45,000 utterances within the entire dataset, a number of different representations are evaluated for their predictive performance. Additionally, ablative evaluations are performed to validate the predictive value of including dialogue data, as compared to an interaction-only baseline approach. During the embedding generation, dialogue is processed using steps such as converting text to lowercase, automatic spell-checking, stemming or lemmatization, and removing stopwords. Due to the size of the different embedding models, dimensionality reduction techniques such as Principal Component Analysis are be investigated. We hypothesize that multimodal student data with dialogue represented using embeddings from newer neural architectures (e.g., BERT, ELMo) will outperform student models induced using interaction-only modalities and dialogue embeddings from older embedding techniques (e.g., GloVe, Word2Vec).

Because there are only 66 students, to obtain enough data for training deep learning models, we generate features in a sequential manner across fixed time intervals. This method allows for post-hoc analysis of a model's predictive accuracy at particular segments of a student's learning session. In addition, the dialogue modality will also be investigated by separating the

student dialogue and tutor dialogue and generating separate embeddings to see if distinguishing dialogue sources provides any improvement in the models' predictive accuracy. The results from this work have the potential to provide valuable information to instructors, course designers, and learning scientists as they can design systems to highlight these particular dialogue patterns to improve user-sensitive features in future learning environments, while also demonstrating an effective approach for harnessing the predictive potential of dialogue data, even when manual annotation or processing is infeasible.

5.1.1 Data Preprocessing and Word Embedding Models

To investigate different representations of student and tutor dialogue, we generated dialogue embeddings using GloVe (Pennington, Socher, & Manning, 2014), BERT (Devlin et al., 2018), ELMo (Peters et al., 2018), and Word2Vec (Mikolov et al., 2013) models; training different non-neural and neural predictive models combining the embeddings with interaction trace data using various multimodal data fusion techniques described in Section 5.1.3. We use BERT and ELMo embeddings because they take into account the contextual placement of a word in a sentence, word order, and also account for out-of-vocabulary words, which distinguish these embedding methods from other word embedding models such as Word2Vec and GloVe, which are used for comparison purposes. Preprocessing steps for the raw dialogue text are described in Section 4.2.5.3. Word embeddings were generated from these four pre-trained models for each word in a sentence, and then the embeddings were averaged across each utterance to obtain a single vector representation.

5.1.2. Affect Model Evaluation

For each approach, we evaluated three multi-class machine learning models: support vector machine (SVM), random forest (RF), Naïve Bayes (NB), logistic regression (LR), and feedforward

neural network (FFNN). Hyperparameter optimization took place for each model. The SVM's hyperparameters were the kernel type and the regularization parameter, *C*. Other tuned hyperparameters include the number of estimators (RF) and regularization penalty (LR). The FFNN was evaluated using three hidden layers of different layer sizes as determined by a random hyperparameter search. The random hyperparameter values were kept consistent across experiments for consistency. Additionally, the learning rate was evaluated as well. For the FFNN, each hidden layer contained a Leaky Rectified Linear Unit as its activation function, with a softmax output layer. The Adam optimization algorithm was used to train the model, with a dropout probability of 0.5 applied before the output layer. Additionally, early stopping was employed during the training process based on a separate validation set, and training was halted using a validation patience of 50 epochs. The model was trained using cross-entropy as the loss function, and training was halted after 1000 epochs if prior convergence was not reached. To obtain a single prediction for a student's learning session, the mean across all predictions was calculated. The models are evaluated using 10-fold nested cross-validation, with 3-fold inner cross-validation used for hyperparameter tuning within each outer cross-validation fold's training set. The cross-validation splits are along student lines to prevent data leakage. A number of different evaluation metrics are used, including Area-Under-Curve (AUC), raw accuracy, and F1 score.

The purpose of these experiments is to investigate whether newer, context-sensitive embedding models such as BERT and ELMo outperform older embedding models such as GloVe and Word2Vec, and whether multimodal data that includes the dialogue modality induces higher performance from the predictive models than interaction-only baselines. In this portion of the work, we use the four aforementioned embedding models, with each model used to generate representations of the dialogue modality that are subsequently fused with the interaction-based

modality. In this way, we aim to demonstrate that the addition of dialogue information can be used to more accurately predict instances of *engagement* and *frustration* in students during the tutoring sessions, and that natural language is a good indicator of affect in learning environments. Additionally, we investigate whether BERT and ELMo embeddings outperform the embeddings generated by Word2Vec and GloVe.

5.1.3 Multimodal Data Fusion

There are multiple approaches to combining the dialogue and interaction modality; we investigate two variations of multimodal data fusion to accomplish this task. "Early Fusion" is feature-level fusion where the dialogue-based PCA components are concatenated with the interaction-level features prior to training a single model. "Late Fusion" is a decision-level fusion that requires a separate model to be trained on each modality independently, with the prediction from each model used to generate a single representative prediction through either average voting or highest-confidence voting. In this case, we use the average of the two predictions from the two modalities to generate a single prediction.

5.1.4 Separate Embeddings

To investigate whether the source of a particular utterance improves predictive performance, we modify the feature engineering process to generate separate embeddings for the dialogue based on whether the utterance originated from the student or the tutor. In this case, across a particular time interval for a single feature vector, all of the embeddings from the student's dialogue were averaged to generate a single embedding vector, and all of the embeddings from the tutor's dialogue were also averaged to generate a separate embedding vector. These two vectors were concatenated prior to training the classifiers, so in this experiment there were twice as many

dialogue features. For these particular evaluations, we treated the student and tutor dialogue embeddings as separate modalities, so the multimodal data fusion techniques (such as Late Fusion) were applied to the two dialogue channels separately, including the Principal Component Analysis.

5.1.5 Results

The predictive results from the different word embeddings are shown in Tables 5.1 and 5.2 for *engagement* and *frustration*, respectively. We use the F1 score as the primary evaluation metric due to its ability to account for both precision and recall of the models and use AUC and Cohen's Kappa as secondary evaluation metrics as they are more informative for imbalanced data. The data in this experiment is relatively balanced due to the use of a median split to generate the classification labels. Results indicate that for both affective states, the BERT embeddings produce the highest predictive performance across all evaluation metrics, with the exception of Kappa for *engagement*. This appears to indicate that BERT embeddings are more predictive of student affect when compared to ELMo, Word2Vec and GloVe embeddings. This may be due to the fact that BERT takes into account the contextual placement of a word in a sentence, word order, and any out-of-vocabulary words. Additionally, it is noted that the inclusion of the dialogue-based modality

Table 5.1. Results for *engagement* prediction.

Modality	Embedding	Model	Fusion Type	F1 Score	Accuracy	AUC	Kappa
Interaction	N/A	FFNN	N/A	0.656	0.568	0.416	-0.008
Multimodal	Word2Vec	SVM	Late Fusion	0.675	0.557	0.506	0.022
Multimodal	GloVe	FFNN	Late Fusion	0.651	0.523	0.486	-0.057
Multimodal	ELMo	FFNN	Early Fusion	0.649	0.518	0.502	0.063
Multimodal	**BERT**	**SVM**	**Late Fusion**	**0.684**	**0.559**	**0.525**	**-0.009**

Table 5.2. Results for *frustration* prediction.

Modality	Embedding	Model	Fusion Type	F1 Score	Accuracy	AUC	Kappa
Interaction	N/A	SVM	N/A	0.639	0.556	0.575	0.093
Multimodal	Word2Vec	FFNN	Early Fusion	0.681	0.534	0.453	0.011
Multimodal	GloVe	FFNN	Late Fusion	0.675	0.535	0.537	0.029
Multimodal	ELMo	FFNN	Early Fusion	0.650	0.506	0.558	0.041
Multimodal	**BERT**	**SVM**	**Early Fusion**	**0.699**	**0.590**	**0.594**	**0.135**

generally led to higher performance in F1 score when compared to an interaction-only model, with the exceptions of GloVe and ELMo embeddings for *engagement*.

It is also noted that ELMo achieved relatively low performance compared to the other embedding baselines. One notable difference in the optimal models for *frustration* and *engagement* is that Early Fusion produced the highest performance for *frustration*, while Late Fusion was the highest performing fusion method for *engagement*. This indicates that the individual modalities for *frustration* did not appear to strongly predict the presence of this affective state when modeled separately, in contrast to the results for *engagement*. This may be due to the fact that modeling dialogue separately from interaction data rather than combining the two modalities allows the modeling framework to extract features from the dialogue that are more predictive of *engagement*, or that the dialogue features alone may be more predictive of *engagement*.

We trained the same classifiers across the same random hyperparameter search grids and report the best results for *engagement* and *frustration* in Tables 5.3 and 5.4, respectively. The first row of each table contains the highest-performing model from Tables 5.1 and 5.2 that combined the student and tutor dialogue. A striking observation from the previous experiment is that treating the student and tutor dialogue as different data channels lead the affect classifiers to perform worse than the optimal models combining the dialogue into a single channel. Based on the results from Tables 5.3 and 5.4, the dual-channel approach underperformed both the multimodal BERT models

Table 5.3. Results for *engagement* prediction (separate embeddings).

Modality	Embedding	Model	Fusion Type	F1 Score	Accuracy	AUC	Kappa
Multimodal	**BERT**	**SVM**	**Late Fusion**	**0.684**	**0.559**	**0.525**	**-0.009**
Multimodal	BERT*	MLP	Late Fusion	0.642	0.505	0.530	0.020

Table 5.4. Results for *frustration* prediction (separate embeddings).

Modality	Embedding	Model	Fusion Type	F1 Score	Accuracy	AUC	Kappa
Multimodal	**BERT**	**SVM**	**Early Fusion**	**0.699**	**0.590**	**0.594**	**0.135**
Multimodal	BERT*	MLP	Late Fusion	0.620	0.503	0.583	0.079

*Indicates separate embeddings for student and tutor dialogue

as well as each of the other non-BERT embedding techniques in terms of F1 Score. This behavior may be due to a number of factors. For example, it was noted that each student sent an average of 33 messages per learning session, compared to each tutor sending an average of 80 dialogue messages per learning session. Each session generated an average of 3.8 features, due to the use of 10-minute time intervals per feature and each session lasting approximately 40 minutes on average. This may lead to an insufficient amount of student-based dialogue compared to the tutor-based dialogue, and since we treat each utterance source separately, this can have an adverse impact on the predictive value of the student-based dialogue channel. In turn, this adversely impacts the predictive accuracy of multimodal data fusion techniques such as Late Fusion. Another possibility is that there may not be enough data samples in the dataset to compensate for the increase in dimensionality in the training data due to the addition of another dialogue channel. Particularly for complex models such as the feedforward neural network, the dataset size (1,324 samples) may not be enough to lead to an increase in performance given the new number of features (7 interaction features + 20 PCA components for the student dialogue + 20 PCA components for the tutor dialogue). Finally, we analyzed the word count for each utterance, and it was observed that a student's word count per utterance was 4.78, while a tutor's word count per utterance was 6.86, which means that a single embedding vector for each tutor represented approximately 43.82% more dialogue compared to the student's vector.

5.2. Misclassification Analysis

The next part of this work investigates the impact of particular dialogue utterances at different time intervals in student learning sessions. For example, it is possible that hints given from the tutor to the student might be more indicative of affect than questions or inquiring remarks from the student to the tutor, or vice versa. To accomplish this task, we analyze each of the students' trajectories

across each time interval and observe the predictions from the optimal model for each affective state. The deltas across each prediction from *t* to *t*+10 where *t* is a given timestamp in minutes were calculate, and a "spike" was determined to be any delta value that was greater than two standard deviations of the predictions for the entire session. We analyzed the dialogue from the tutor and the student that occurred during the ten-minute segment that caused a significant increase or decrease in the model's prediction. In this work, we focus primarily on false positives and false negatives to observe any dialogue patterns that tend to cause a classifier to misclassify the affective states of each student. A visualization of a student's trajectory with the model's affective state prediction, ground truth label, and highlighted "spike" are shown in Figure 5.1, which depicts a false negative prediction from the model, which was adversely impacted by the features in the first 10 minutes.

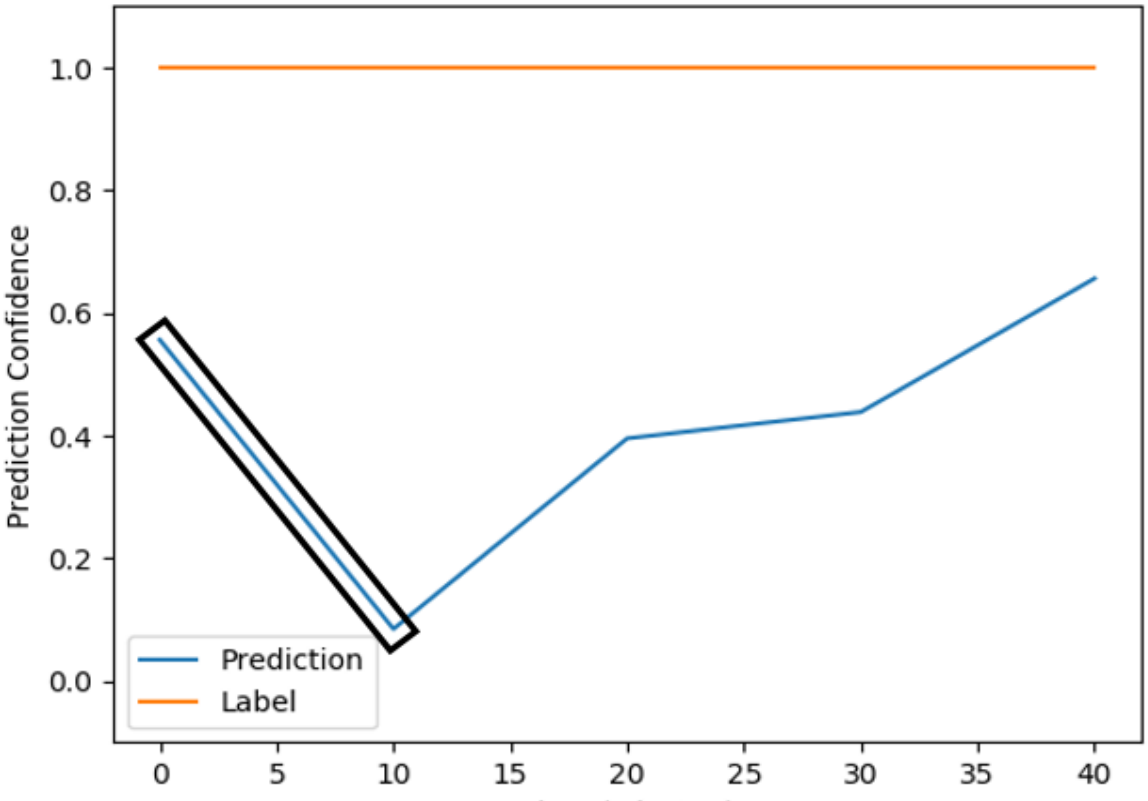

Figure 5.1. Example of misclassification "spike" across a student's learning session.

Because of the large amount of dialogue data within this corpus, we used the Linguistic Inquiry and Word Count (LIWC) application to perform an automated text analysis on the individual utterances (Pennebaker et al., 2015). LIWC, which is often employed to analyze textual features of dialogue, utilizes a series of word vocabularies that are associated with cognitive, social, and affective sentiments within text-based discourse. This allowed for a more empirical evaluation of the different dialogue patterns across the individual time intervals described in Section 4.2.5. We investigated the LIWC features that experienced the largest changes in value during time intervals where a classifier produced a misclassification "spike." These features are shown for student dialogue, tutor dialogue, and the combined dialogue in Table 5.5 below. Negative numbers indicate a decrease in average values from spike intervals to non-spike intervals, and LIWC features are within a 0-100 range. There were no instances of false negative "spikes" for *engagement*.

5.2.1 Results

As a result of the student-tutor utterance imbalance described in Section 5.1.5, LIWC features varied more for tutor-only and combined dialogue, with 10 of the 15 listed features from the combined dialogue also belonging to the top 5 most varying tutor-only features. As a general observation, it was noted that the LIWC features typically decreased during false positive "spikes," while the LIWC features typically increased during false negative "spikes." We also noted that the LIWC features generated from student-only dialogue contained the highest variance, most likely due to the lower sample size of utterances from this group.

Table 5.5. LIWC features (Pennebaker et al., 2015) with highest shifts during misclassification "spike" intervals.

Engagement (False Positive)

Combined		Student-Only		Tutor-Only	
Authentic	-2.066	netspeak	-2.449	Authentic	-2.267
Exclam	1.464	affect	1.736	Exclam	1.857
Period	-1.213	number	-1.695	Period	-1.551
QMark	-0.746	Dic	1.467	Dic	-1.154
auxverb	-0.637	Authentic	-1.433	QMark	-1.069

Frustration (False Positive)

Combined		Student-Only		Tutor-Only	
Analytic	-2.432	affect	-3.221	Analytic	-2.690
AllPunc	1.785	AllPunc	2.736	informal	-1.614
Authentic	-1.498	posemo	-2.703	Clout	-1.477
informal	-1.104	Authentic	-2.278	Authentic	-1.285
reward	0.874	Exclam	2.016	assent	-1.204

Frustration (False Negative)

Combined		Student-Only		Tutor-Only	
Tone	12.865	Tone	16.721	Tone	11.190
Dic	7.232	Dic	8.982	Clout	7.263
Clout	6.031	AllPunc	-8.230	Dic	6.456
verb	4.754	verb	7.652	drives	5.304
posemo	4.590	posemo	7.620	reward	4.950

Tutor dialogue appeared to have the strongest predictive impact on the classifier performance for false predictions of *engagement*. Notably, tutor "authenticity" (self-referencing) decreases and "exclamations" (exclamation marks) increase during inaccurate prediction trends towards *engagement*. While "exclamations" have been previously shown to correspond to heightened affect intensity, this increase is valence-agnostic (Hutto & Gilbert, 2014). More importantly, many pre-trained word embedding models, including BERT, do not account for punctuation. A decrease in "netspeak" (informal language such as "lol" or "btw") was also detected when misclassifying *engagement*, and the presence of "netspeak" may not be accurately determined by the selected word embedding model (BERT) due to being labeled as being "out-of-

vocabulary" potentially misleading the affect classifier. An increase in "affect" (indications of affective processes) was also attributed to false predictions of *engagement*, and it should be noted that this LIWC feature encompasses both negative and positive valenced affect, not just positively-valenced states such as *engagement*. The same aspect of this LIWC feature also appears to impact false positive identifications of *frustration*, based on decreases in "affect" in student-only tutorial dialogue. Positive emotion ("posemo") also decreases significantly during false predictions of student *frustration*, understandably leading to inaccurate model performance. While this decrease may indeed correspond to momentary instances of *frustration*, the affect labels represent affect over the entire learning session, and this discrepancy in granularity may have a role in such misclassifications. "Tone" refers to high levels of positive and negative valence in the utterances and was by far the most impactful LIWC feature during false negatives of *frustration*, however, it appears that the affect model encounters difficulty correctly correlating the "Tone" of the dialogue with positive or negative valence, leading to inaccurate identifications of *frustration*. It was noted that all punctuation ("AllPunc") experienced changes during misclassification intervals, indicating a potential disadvantage of using word embedding models that do not consider punctuation when generating latent representations of natural language. Finally, it was observed that positive emotion increased in students' dialogue during false negatives of *frustration*, leading to understandable, albeit inaccurate, predictions of non-frustration in students. While the models were not explicitly trained on LIWC features, this analysis points to potential sources of inaccuracy in these affect models. It also highlights an inverse relationship between the presence of *frustration* and positive sentiment in dialogue, potentially resulting from the granularity discrepancy between post-surveys and the prediction intervals as mentioned before.

A particular challenge with this exercise was the preprocessing of the dialogue prior to generating the text embeddings, particularly the removal of punctuation and lemmatization. Because *JavaTutor* dialogue centers on the Java programming language, symbols such as brackets and parentheses commonly occur within the dialogue and often provide insightful value to the predictive models but are removed during preprocessing. Additionally, student dialogue often consists of short or one-word answers and is often either removed or shortened further so that the resulting preprocessed dialogue is very frequent and not predictive for the affect classifier models. As noted above, another challenge with this work is the imbalance of student and tutor dialogue. We observe the potential impact that this may have when separating the tutor and student dialogue into separate channels, but because the overall work focuses on only predicting *student* affect, this may contribute greatly to the overall performance of the affect detection models. A final challenge was handling out-of-vocabulary words, which often occurred with words specific to the *JavaTutor* lesson or the Java programming language. However, this issue occurred primarily with Word2Vec and GloVe embeddings, but more sophisticated models such as BERT are designed to handle this issue and the use of BERT helped to overcome this issue.

Dialogue in learning environments has received increasing attention due to its predictive potential for student affect modeling, but effectively representing dialogue while retaining its predictive value raises questions about scalability and generalizability. While latent representations of dialogue such as word embeddings may address these issues, they often lack the interpretability necessary for uncovering dialogue patterns and features that cause misclassifications by affect models. We addressed this issue by investigating different word embedding representations for training student affect models while also sequentially evaluating different dialogue patterns corresponding to misclassifications in two affective modeling tasks.

We observed that the use of a more complex, context-sensitive embedding model, BERT, on student-tutor dialogue produced higher predictive performance than Word2Vec, GloVe, and ELMo embeddings for two affect detection tasks. Additionally, we observed that separating the student and tutor dialogue into different data channels does not improve the predictive performance, potentially due to data sparsity or dialogue imbalances between students and tutors. Finally, we performed a post-hoc analysis of various discourse patterns with LIWC-based features that may point to causes for misclassifications, including short utterances from students that are not strongly predictive, omission of potentially predictive natural language features (e.g., punctuation), and out-of-vocabulary dialogue. Ultimately, our work demonstrated the effectiveness of an affect detection framework that utilizes dialogue embeddings in an interactive learning environment and avoids labor-intensive and non-scalable methods such as manual natural language annotation.

5.2.2 Limitations

There are several limitations to this work that warrant further discussion. On a broad scale, affect detection is a particular challenging task, as individuals often exhibit different emotions through different physical behavioral cues. In particular, accurately detecting affective states such as *engagement* and *frustration* using only interaction and dialogue data is particularly challenging, as certain behaviors such as disengagement or non-*frustration* are often not adequately indicated by individuals through interaction and dialogue data. For example, an absence of dialogue data may be indicative of disengagement but may also likely be exhibited by an individual that is highly engaged with a programming task that doesn't require the aid of the tutor. Likewise, it is possible that the mere presence of dialogue from the student is predictive of *engagement* during the learning session. Additionally, asking the student to self-report a single affective state over the course of an

hour-long learning session is not granular enough to accurately capture short-term emotions such as *frustration*. In this particular setting, because of the tutor-student dynamic in the *JavaTutor* environment, it is likely that the students exhibited more self-restraint than in more realistic scenarios, and the utterances from the students may be more withdrawn or less expressive.

Additionally, the nature of the dialogue between the tutor and the student may also be a limiting factor in the performance and implementation of the affect models. For example, it was observed that there was a significant imbalance between the number of student utterances and the number of tutor utterances, with tutors often sending three times as many dialogue messages as students. Additionally, the students' utterances often have half as many words on average relative to tutors' utterances, which could adversely impact the predictive value of the dialogue modality, especially considering that for the first set of experiments, the dialogue embeddings were averaged together regardless of whether they originated from a student or a tutor. It was also observed that many of the students' utterances were one-word replies to tutor questions that often only required a "yes" or "no" response, which are very common words and likely not holding much predictive value.

While the use of embeddings is much more scalable than other approaches to handling dialogue data such as manual annotation, these word embeddings from models such as BERT or ELMo are latent representations of natural language and are not as interpretable as other dialogue representations. This is further complicated by the fact that the embeddings have high dimensionality and are generated on a word level. Because the dialogue consists of sentences and sometimes paragraphs of words in a single message, we average across all embeddings in a single utterance. Additionally, to generate a single feature for a 10-minute interval due to the sequential nature of our data, we average again across multiple sentence embeddings. This is a simplistic

approach that may result in each word embedding's predictive value becoming lost in the multiple averaging steps, and it also makes interpreting the dialogue's predictive components more difficult. Additionally, the analysis of the word embeddings is made more complicated when other dimensionality reduction such as Principal Component Analysis takes place. This is a common issue with the use of text embeddings and is a tradeoff that must be acknowledged with many different natural language processing tasks.

Another limiting factor of this particular project is the dataset size. Although the number of feature vectors can be increased by decreasing the time interval of the feature engineering, because of the relatively infrequent nature of the dialogue utterances and the interaction features, this often results in repetitive or otherwise redundant features that do not improve the performance of the machine learning models. Additionally, the relatively low number of data points is inhibitive to the use of sequential deep learning models often used in similar natural language processing tasks, such as Long Short-Term Memory (LSTM) networks. As a result, we notice that the performance of the affect models may be dependent on the sampling rate of the dataset, which is influenced by the frequency of the dialogue, which can often not be pre-determined.

5.2.3 Implementation Prerequisites

A benefit to our proposed multimodal framework is that it makes use of two unintrusive modalities that do not require any external sensors or hardware, and do not encounter common issues with sensor-based data capture such as noise, mistracking, or miscalibration. Additionally, our proposed approach makes use of commonly available embedding techniques such as BERT. The affect models evaluated in this project are also common classical machine learning models such as Support Vector Machine, Random Forest, and Feed-Forward Neural Networks. The primary prerequisite to implement our findings in this work is a text-based dialogue channel and interaction

trace data captured by any software framework. However, it should be noted that the features used in our specific work were relatively specific to the *JavaTutor* platform, such as the number of student/tutor messages or successful compiles, and the work that focused on separating the student and tutor dialogue requires that the dialogue originate from at least two different sources. However, the classifier models and the dimensionality-reduction are relatively domain-agnostic. Additionally, it should be noted that another prerequisite for implementing our findings is the use of sequential data, as all of our results and the post-hoc analyses are centered around the use of multiple predictions across a given time sequence. Ultimately, our findings are primarily implications on the predictive value of dialogue-based information as well as the benefits of using more complex, context-sensitive embeddings techniques, along with the potential adverse impact of issues common to datasets within the education domain, such as sparse or infrequent discourse data, as well as issues related to natural language or domain-specific language that may lead to misclassification in tasks related to affect or emotion detection.

CHAPTER 6

ENHANCING MULTIMODAL AFFECT DETECTION: POSTURE + EDA

The work described in this chapter addresses the affect detection models' predictive performance using the USMA multimodal dataset described in Chapter 4. The first experiment explores the performance of various neural and non-neural predictive models using the two sensor-based modalities: posture and electrodermal activity (EDA). Additionally, this experiment investigates feature-level and decision-level data fusion and their respective impacts on the overall performance. The results from this chapter serve as a baseline to motivate the multimodal data imputation of the EDA modality in Chapter 7. The methods described in this Chapter are implemented in a similar manner in Chapter 8 and is validated using the *JavaTutor* dataset in that chapter, but we do not perform a similar validation task in this chapter because the *JavaTutor* dataset only has one sensor-based modality available.

We first investigate the effectiveness of sensor-based multimodal models for affect detection using student posture information captured by the Microsoft Kinect, as well as EDA data captured by the Affectiva Q-Sensor. In this section, we describe the implementation and performance of an end-to-end data processing pipeline for the posture- and EDA-based modalities captured from the students' interactions with *TC3Sim*; and we provide additional detail into particular aspects of the pipeline including the data preprocessing, feature selection, affect modeling, and multimodal data fusion.

6.1 Multimodal Affect Detection Performance

In this section, we evaluate an affect modeling pipeline to determine the performance of unimodal and multimodal models. We provide additional detail on the construction of the multimodal

datasets, including preprocessing techniques such as data upsampling and feature selection. We evaluate the performance of support vector machines (SVM) and deep feedforward neural networks as affect classifiers using unimodal data, in addition to multimodal data comprised of posture and EDA data. Finally, we evaluate three variations of data fusion for the multimodal affect classifiers. Results suggest improved performance of multimodal classifiers compared to unimodal classifiers trained on separate posture- and EDA-based modalities, and they reveal the impact that different data fusion techniques have on the accuracy of multimodal affect models. Additionally, the preprocessing techniques described in this section are also implemented in a majority of the subsequent work related to this particular dataset (USMA dataset).

6.2 Multimodal Datasets (USMA)

The primary goal of this work is to demonstrate the effectiveness of a multimodal classification system for affect detection using two modalities: Kinect-based posture data and Q-Sensor-based EDA data. To ensure that both modalities are present in each data sample, any BROMP observation with missing or invalid EDA data was removed from the dataset. The Q-Sensor experienced frequent stops in data logging throughout the initial data collection. This issue resulted in 333 BROMP observations missing EDA information, while a subset of 422 data samples contained both the posture and the EDA modalities. Therefore, the classifiers were trained on a dataset using 422 BROMP observations containing correlated posture and EDA data. After the aforementioned BROMP observations were removed from the dataset, five separate datasets were created with each dataset containing binary labels indicating the presence of a particular affective state (*bored*, *confused*, *engaged concentration*, *frustrated*, and *surprised*). Each dataset was scaled using z-score standardization. This method ensures that each attribute of the feature vectors has the same mean and standard deviation but allows for different ranges.

6.3 Dataset Upsampling

Because of the class imbalance present in several of the datasets, particularly *frustrated* and *surprised,* we perform class upsampling for each dataset to bring the class distribution to a more uniform level. In this work, the upsampling was accomplished using a minority class cloning technique. This approach involves duplicating each data sample belonging to the minority class according to the ratio of the majority to minority class labels for each individual affective state. This produces five distinct upsampled datasets corresponding to the five affective states.

6.4 Forward Feature Selection (USMA)

Prior to training the classifiers, each dataset underwent forward selection for the purpose of feature selection. Forward feature selection involves iterating through combinations of features in a greedy fashion, beginning with feature vectors of size 1 and continuing until *k* features are selected or all combinations of features are exhausted. For our work, a *k* value of 10 was chosen. The model used in feature selection was the sequential minimal optimization (SMO) support vector machine (Platt, 1998). This polynomial-kernel model was selected due to its linear memory requirements and scalability, as a high number of models were trained to obtain the "optimal" features. A feature is selected as "optimal" if its addition to a feature set yields improved accuracy for the SMO-SVM model as measured by Cohen's Kappa. The feature selection was implemented using RapidMiner 9.0 (Mierswa et al., 2006). This platform was selected due to its convenience as a toolkit for implementing the data processing pipeline, as well as its prior use in affect detection tasks.

6.5 Multimodal Data Fusion

To evaluate different methods of integrating the two modalities for affect classification, we implement several variations of multimodal data fusion. We test two types of data fusion: feature-

level fusion ("Early Fusion") and decision-level fusion ("Late Fusion"). Early Fusion involves the concatenation of features from the posture and EDA modalities prior to training the affect classifier. Late Fusion calls for the training of separate classifiers for each modality, and the predicted confidence levels of each binary class (positive or negative label of affective state) are processed by a voting schematic to produce a singular prediction of the affective state. The voting schematic can be implemented in different ways, such as majority voting, averaging, or weighting (Baltrušaitis et al., 2019). For this work, we take the highest confidence value across the two classifiers and use the associated class as our final representative prediction. Two different variations of Early Fusion are also evaluated. The first variation, referred to in this work as "Early Fusion 1", concatenates the features prior to the feature selection process. The other variation, referred to as "Early Fusion 2", performs feature selection on the separate modalities, and only the selected features from each modality are concatenated prior to training the classifiers. A visualization of each data fusion technique using the posture and EDA modalities is shown in Figure 6.1.

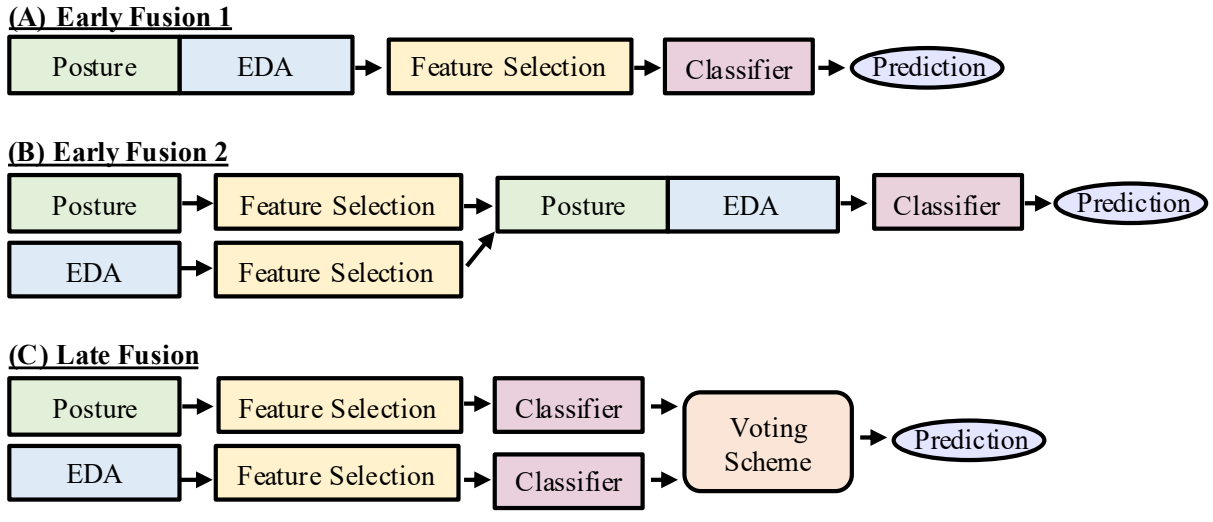

Figure 6.1. Early Fusion 1, Early Fusion 2, and Late Fusion.

6.6 Model Evaluation

Prior work has demonstrated the effectiveness of deep neural networks in affect classification tasks (Henderson, Rowe, et al., 2019). We utilize the same neural network approach using FFNNs and compare it with SVM models. The SVMs contain a radial kernel function with a convergence epsilon of 0.001 for a maximum of 100,000 iterations. The FFNN architecture contained five feed-forward layers and a binary classification layer. Each layer's activation function was a Rectified Linear Unit (ReLU). Each network was trained for ten epochs with the ADADELTA adaptive learning rate (Zeiler, 2012). A separate classifier was trained for each affective state, using the selected features of the oversampled data as described in Section 6.3.

The classifiers were evaluated using 10-fold cross validation, with the data split on a student level to ensure that all data from individual training sessions were kept in the same fold. The same batches of data were maintained across all modeling approaches to ensure fair comparisons across classifiers. The unimodal baseline classifiers and Early Fusion pipelines were implemented using RapidMiner 9.0. RapidMiner does not support decision-level fusion, so the Late Fusion pipeline was implemented using Python 3.6, while the classifiers were still implemented in RapidMiner.

6.7 Results

Unimodal classifiers were trained on the posture and EDA modalities independently to provide a baseline for the multimodal classifiers' performance. The results for the posture and for the EDA-based unimodal classifiers for each affective state are shown in Table 6.1. Evaluation metrics include Cohen's Kappa, raw accuracy, and F1 Score. Particular focus is given to Cohen's Kappa because of its ability to account for the possibility of correct classification due to random chance. The posture-based SVM returned the highest Kappa for four of the five affective states, and the

EDA-based SVM outperformed the FFNN for three of the five affective states. The FFNN model performed poorly on a majority of the evaluations, returning a negative Kappa on two of the posture-based states and four of the five EDA-based states. This indicates that the FFNN is no better than a random classifier for a majority of states.

Table 6.1. Unimodal classifier performance for affective states.

Bored

	Posture			EDA		
Classifier	**Kappa**	**Accuracy**	**F1 Score**	**Kappa**	**Accuracy**	**F1 Score**
SVM	**0.004**	0.607	0.013	**-0.042**	0.500	0.286
FFNN	-0.001	0.408	0.530	-0.047	0.360	0.478

Confused

	Posture			EDA		
Classifier	**Kappa**	**Accuracy**	**F1 Score**	**Kappa**	**Accuracy**	**F1 Score**
SVM	**0.002**	0.566	0.040	**0.033**	0.533	0.319
FFNN	-0.003	0.566	0.040	-0.083	0.387	0.529

Engaged Concentration

	Posture			EDA		
Classifier	**Kappa**	**Accuracy**	**F1 Score**	**Kappa**	**Accuracy**	**F1 Score**
SVM	**0.065**	0.484	0.523	-0.108	0.449	0.437
FFNN	0.020	0.484	0.523	**-0.013**	0.541	0.682

Frustrated

	Posture			EDA		
Classifier	**Kappa**	**Accuracy**	**F1 Score**	**Kappa**	**Accuracy**	**F1 Score**
SVM	**0.092**	0.553	0.441	-0.046	0.491	0.539
FFNN	0.063	0.501	0.650	**0.011**	0.387	0.641

Surprised

	Posture			EDA		
Classifier	**Kappa**	**Accuracy**	**F1 Score**	**Kappa**	**Accuracy**	**F1 Score**
SVM	-0.236	0.632	0.040	**0.086**	0.607	0.357
FFNN	**0.020**	0.270	0.431	-0.001	0.222	0.478

The posture classifiers performed relatively poorly on *bored, confused,* and *surprised*. It is worth noting that *surprised* contains the lowest number of positive instances within the dataset, which may contribute to the poor performance. Additionally, it is possible that postural behavior may not distinguishably change between positive instances of *bored* and *confused*, leading to common misclassifications across the two states. The EDA classifiers performed poorly on the affective states of *bored, engaged concentration,* and *frustrated*. It is noted that the EDA modality contains significantly fewer features than the posture modality, and this may have caused additional misclassifications due to the lack of substantial training data available. It is also possible that the EDA modality may not contain enough variance for the classifiers to distinguish between positive and negative instances of affective states. Additionally, the EDA classifiers face the task of distinguishing between different changes in the EDA measurements, and determining whether such changes can be attributed to a particular affective state or to another cause. However, this proves to be more difficult than using the posture modality due to the singular dimensionality of the EDA channel. To further illustrate this issue, a graphical representation of the change in EDA throughout a single student's session is shown in Figure 6.2.

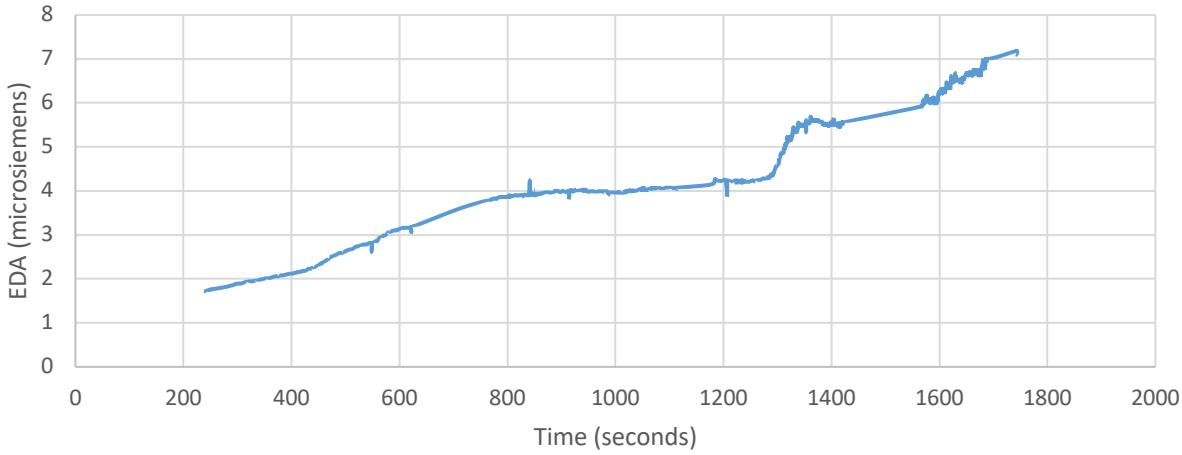

Figure 6.2. EDA trajectory measured over a single learning session.

The SVM was selected as the classifier used to implement and evaluate the data fusion methods discussed in Section 6.5. The feature selection algorithm and classifier configuration were the same as those used in the unimodal approach, and the same student-level groupings were maintained. The three data fusion approaches were evaluated for each affective state, and the results for each state are shown in Table 6.2. Early Fusion 2 returned the highest Kappa for *bored, engaged concentration,* and *frustrated*. Early Fusion 1 returned the highest value for *confused*,

Table 6.2. Performance of Early Fusion 1, Early Fusion 2, and Late Fusion for affective states using SVM models.

Bored			
Classifier	**Kappa**	**Accuracy**	**F1 Score**
Early Fusion 1	-0.082	0.466	0.164
Early Fusion 2	**0.041**	0.532	0.356
Late Fusion	-0.056	0.583	0.145
Confused			
Classifier	**Kappa**	**Accuracy**	**F1 Score**
Early Fusion 1	**0.049**	0.566	0.300
Early Fusion 2	-0.004	0.515	0.321
Late Fusion	0.032	0.597	0.148
Engaged Concentration			
Classifier	**Kappa**	**Accuracy**	**F1 Score**
Early Fusion 1	-0.064	0.446	0.393
Early Fusion 2	**0.068**	0.542	0.491
Late Fusion	-0.035	0.481	0.459
Frustrated			
Classifier	**Kappa**	**Accuracy**	**F1 Score**
Early Fusion 1	0.191	0.657	0.656
Early Fusion 2	**0.246**	0.594	0.483
Late Fusion	0.119	0.568	0.490
Surprised			
Classifier	**Kappa**	**Accuracy**	**F1 Score**
Early Fusion 1	**-0.021**	0.590	0.053
Early Fusion 2	0.013	0.682	0.080
Late Fusion	-0.192	0.514	0.124

while the EDA baseline was the highest value for *surprised*. One possible reason that Early Fusion 2 is the highest-performing data fusion method is that feature selection is performed separately on each modality prior to concatenation for inducing each classifier. This means that if each feature selection algorithm selects up to the *k*th best features, then the combined feature vector can contain up to 2*k* features, twice as many features as allowed by Early Fusion 1. This increase in features may boost the performance of the classifier. Late Fusion can also work with 2*k* features, but the features are split between the two unimodal classifiers before decision-level fusion. Early Fusion 2 also explores the correlations between various inter-modal attributes more deeply compared to Early Fusion 1. The complex relationships between various intra-modal features are explicitly modeled in the feature selection performed on each independent modality, while the correlations between the selected inter-modal features are explored when training the primary classifier following feature selection. However, these two stages are performed simultaneously in Early Fusion 1, so certain complex relationships may not be detected.

Late Fusion provides the ability to "correct" a possibly incorrect prediction across the two modalities. For example, if the postural classifier produces an incorrect prediction of "true" with a confidence level of 0.6, but the EDA classifier produces an accurate prediction of "false" with a confidence level of 0.8, then the EDA modality overrides the incorrect prediction because of the selected voting schematic. However, Late Fusion was not the optimal fusion method for any of the affective states, though its effectiveness as a multimodal fusion technique has been demonstrated in other affective computing tasks (Henderson, Rowe, et al., 2019).

Of note is the performance of the multimodal classifier on the *frustrated* dataset compared to the other affective states, as the classifier achieved substantially higher Kappa scores. One possible explanation for this behavior is that negative, high-arousal emotions such as *frustration*

or anger have been shown to occur relatively infrequently in students engaged with computer-based learning environments (Harley et al., 2015). This may mean that the recorded instances of *frustrated* contain more distinguishable features than those contained by other common, low-arousal affective states such as *bored* and *engaged concentration*, encouraging higher performance from the frustration-based classifier. Additionally, *frustrated* has been demonstrated to elicit higher EDA levels (Ramachandran et al., 2017), indicating that the inclusion of the EDA modality with the posture modality provides additional informative features to the feature vectors, contributing to the relatively high performance of the classifier.

Although the multimodal classifiers generally outperformed unimodal classifiers, the highest-performing model returned a relatively low Kappa compared to the performance of a human BROMP labeler (~0.6). However, this threshold can vary depending on the affective state and intervention associated with each state. For example, identifying instances of *engaged concentration* can be viewed as a lower priority than identifying instances of *frustrated* or *bored*, as these affective states often necessitate a dynamic intervention to improve learning outcomes. However, the Kappa values for most of the classifiers fall below 0.05, indicating significant difficulty for several classifiers in achieving consistent performance across multiple affective states.

Previous research efforts have demonstrated that the EDA modality does not have a tightly coupled relationship with different affective states when compared to other higher-dimensionality modalities such as facial expression and gesture (Harley et al., 2015). The results of our work also indicate that the EDA modality resulted in at least one classifier returning a negative Kappa for all five affective states. Possible explanations for this behavior include an inadequate amount of training data, lack of variance or distinguishable trends across the observed time windows, or lack

of predictive features. However, our results indicate that the EDA modality does generally improve classifier performance when used in conjunction with the posture modality, and the current data representation of this data source may not be adequately predictive to be a reliable inclusion in multimodal affect detection frameworks.

CHAPTER 7

AUTOENCODER-BASED MULTIMODAL DATA IMPUTATION

As mentioned in Section 6.2, the Q-Sensor experienced issues with inconsistent data logging throughout the data collection, which resulted in many BROMP observations containing missing EDA information. Specifically, 333 BROMP observations were missing EDA information, while the remaining 422 data samples contained both the posture and the EDA information. This problem affected approximately 44.1% of the entire dataset. A common approach to missing or noisy data in multimodal data collections is to simply remove the data samples that are corrupted, which is the approach taken in Chapter 6. However, this can result in a substantial reduction in the amount of data available to induce various machine learning models and can have an adverse impact on the predictive performance as a result. Based on the results in Section 6.7, we hypothesized that the lack of training data impacted the performance of the predictive affect models, particularly the deep neural networks. As a means to potentially improve the performance of the affect models, we seek to enable the corrupted multimodal data to be used for training the models through missing data imputation. By "imputing" (i.e., replacing) the missing EDA values, this allows for the entire posture-based corpus to be used to train the affect models and can lead to enhanced predictive performance.

To this end, we propose a framework for addressing missing data in our multimodal affect detection task. We trained an autoencoder neural network with the subset of data containing all modalities. The autoencoder was trained to reconstruct the original complete dataset using artificial noise injection, thus simulating the missing modalities. Subsequently, the trained autoencoder was used to impute the missing EDA values. We also investigated the use of the encoder component of the autoencoder as an alternative to dimensionality reduction during the feature extraction

process. Afterward, we identified the highest-performing classifier for each of five affective states, comparing several classifiers trained on the reconstructed multimodal dataset using the data fusion techniques described in Section 6.5. Results indicate that autoencoder-based data reconstruction outperformed other baseline data imputation methods based on classifier performance, and multimodal affect detection yielded improved classifier performance compared to unimodal affect detection. To demonstrate the generalizability of this approach (and subsequent deep learning-based techniques for multimodal affect detection), we also evaluate the same approach using another multimodal affect-oriented dataset, the *JavaTutor* dataset.

7.1 Multimodal Stacked Denoising Autoencoders

In this work, we handled missing data by employing a specialized variation of a denoising autoencoder that is based on Multimodal Autoencoders (MMAEs) (Vincent et al., 2010). The model used in this process involved feature-level fusion (i.e., Early Fusion 1) of the modalities, which were subsequently used to train an autoencoder model. The model was trained to reconstruct the original dataset by converting the dataset to a latent representation following artificial noise injection on select modalities. This process was initiated by taking the complete multimodal dataset and normalizing all features to be in the range [0, 1]. Before propagating the complete dataset through the autoencoder, each multimodal data sample was injected with two types of noise: a simple masking noise and a complete removal of one modality. For the simple masking noise, 5% of the features for the observation were randomly selected and these values were set to 0. The removal of the modalities was performed by randomly selecting a modality and setting each feature within this modality to -1. The MMAE was then trained to reconstruct the original data by using this compromised dataset. This process enabled the autoencoder to more accurately

reproduce a full multimodal dataset when faced with missing or invalid data within certain modalities.

In a similar manner to the training data, all features of the full dataset were normalized to be in the range [0,1]. The values that were missing or invalid, which included blank cells, unique identifiers, and other representations, were all set to -1. Once the MMAE was trained on the corrupted multimodal dataset, the original dataset, including missing data, was passed through the autoencoder. The output of the autoencoder's decoder component included imputed values for all observations. Instead of using the latent data representations from the autoencoder's encoder component as the input to classification algorithms, as in (Jaques et al., 2017), we propagated the full set of observations through the trained network, including the decoder component of the MMAE, resulting in an output of the same dimensionality as the input. As a result, the output then contained imputed values for all original values in the dataset. Because the decoded, imputed dataset was the same dimensionality as the original dataset, any imputed data values that contained a corresponding intact data value was replaced with this value. This ensures that only missing values were replaced with imputed values, and pre-existing values (e.g., posture feature vectors) were retained. A visual representation of this process is found in Figure 7.1, for a dataset containing *m* attributes and *n* data samples.

This approach confers a significant advantage over using the latent representation of the data: it yields an interpretable set of features. Performing feature selection on this imputed data can then generate a more human-understandable set of features, as opposed to the result of the latent data representation from the encoder alone. In addition, it also affords the ability to use other dimensionality reduction techniques such as PCA or even another neural network architecture to find more compact representations of the data and offers more flexibility for interpretable

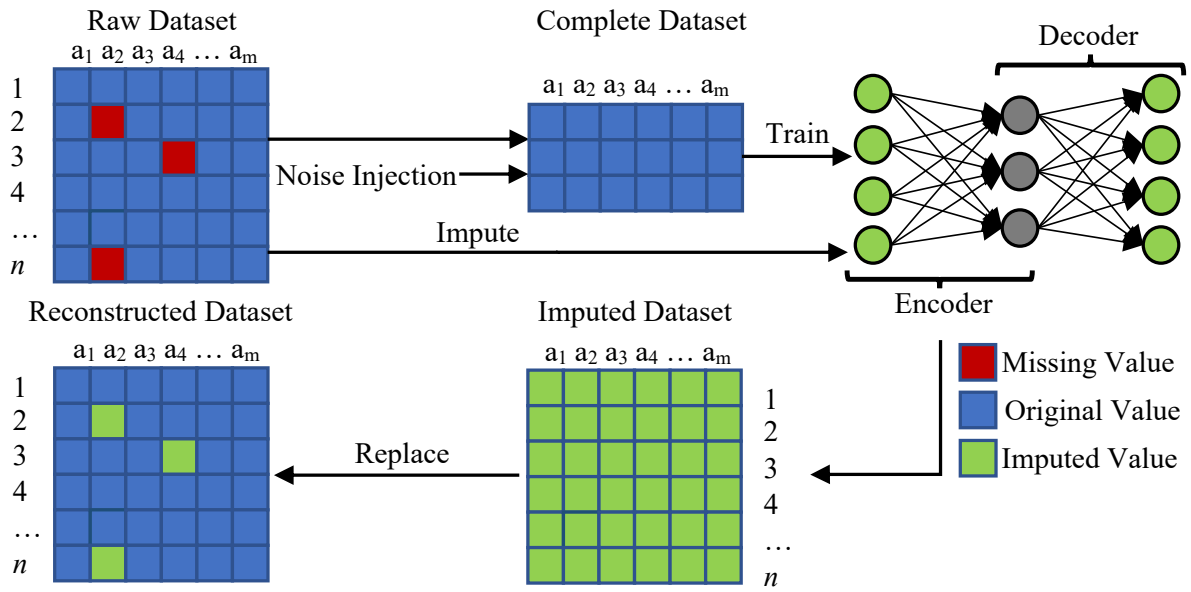

Figure 7.1 Multimodal data imputation using the MMAE process.

preprocessing or feature-selection techniques following the multimodal data imputation within the affect detection pipeline.

7.2 Model Evaluation (USMA)

The autoencoder used to perform this imputation contained a single hidden layer with 30 nodes, which performed well in terms of input reconstruction. The layers within the autoencoder utilized a sigmoidal activation function due to the normalization of the training data. We trained the model using the ADADELTA optimization method (Zeiler, 2012), using mean squared error as the cost function. The training and forward propagation of the missing data was performed with the Keras deep learning toolkit with a TensorFlow backend.

Following the data imputation process, significant class imbalances were still present in several of the affective states. To resolve this issue prior to training the affect detection models, the minority class in each binary affect dataset was upsampled using a minority cloning technique.

This upsampling technique was selected due to its effectiveness in prior work, as discussed in Section 6.3.

A data processing pipeline was established with the objective of investigating a set of classifiers to determine the optimal affect detection model for each individual affective state. Because of the high number of available features within each multimodal dataset, principal component analysis (PCA) was used to reduce the number of dimensions for each classification task. PCA helps remove noise from a high dimensionality space relative to the size of the data, and also accounts for potential multicollinearity among features. For fair comparison in our experiments, we used PCA to transform the data to the same dimensionality as the latent data output of the autoencoder's hidden layer (30). This process ensured that the new orthogonal features were comparable to the latent features produced by the encoder portion of the autoencoder.

To determine the optimal classifier for each affective state, we investigated five different model types: SVM, J-48 decision tree, J-Rip propositional rule learner, logistic regression, and deep FFNN. Each classifier was trained using student-level 10-fold cross-validation, ensuring that data from a single student's session was never split across both the training and test sets, to avoid positively biased results. Positive instances of emotions generated by the prior upsampling process were also removed from the test set, to avoid inflated results as well.

After the optimal classifier for each affective state was determined, we evaluated different variations of multimodal data fusion to determine if feature-level fusion or decision-level fusion enhanced the performance of the classifier. The multimodal data fusion methods are analogous to those described in Section 6.5, although we use PCA as the dimensionality reduction technique instead of forward feature selection. Early Fusion 1 involves concatenating the posture and EDA features prior to the PCA dimensionality reduction and training a single classifier on a dataset

containing *k* attributes. Early Fusion 2 uses PCA dimensionality reduction on each individual modality, producing two individual datasets, each consisting of *k/2* attributes. These attributes are then concatenated in a similar fashion to Early Fusion 1, and subsequently used to train a single classifier.

Late Fusion involves performing PCA dimensionality reduction on two separate modalities with each resulting data channel containing *k* attributes each. Each data channel is then used to train two separate unimodal classifiers. The output of each classifier is a two-element confidence vector representing whether a certain data sample contains a positive or negative instance of the target affective state. A voting schematic is then used to determine the overall representative prediction of the data fusion system.

We experimented with two different voting schematics: highest confidence level, and highest *average* confidence level. For the former, the class with the highest confidence level was selected as the prediction. For the latter, the confidence levels for each class were averaged across each classifier, and the class with the highest average was selected as the prediction. This data pipeline was implemented and evaluated using RapidMiner 9.0 (Mierswa et al., 2006), while the data filtering and distillation, noise injection, and data imputation were performed using Python 3.

Finally, we evaluated classifiers trained on the latent data representations produced by the encoder component of the autoencoder (Jaques et al., 2017) against the classifiers trained on the decoded, reconstructed data. The final results from the multimodal classifiers were then compared with unimodal classifiers trained solely on the posture data to determine whether the addition of the EDA modality through data imputation improved affect detector performance. This allows us to determine whether the use of the multimodal autoencoder-based imputation is justified through the improvement of the affect classifiers using multimodal data versus unimodal data.

7.3 Results (USMA)

We compare our method of data imputation to mean imputation, a commonly used approach that imputes missing data points using the mean of the available data for a given feature (Weiss et al., 2016). Two separate SVM models were trained, one model on a reconstructed dataset using the autoencoder-based imputation, and the other model on a dataset constructed using mean imputation. Cohen's Kappa was used as the primary evaluation metric for classifier performance due to its ability to determine a classifier's ability to perform at a higher success rate than chance. The multimodal dataset was comprised of vectors that contain posture and EDA data concatenated at a feature level. Table 7.1 shows a comparison of Kappa values for each trial with each affective state, with the best imputation method for each classifier shown in bold. Results indicate that autoencoder-based data imputation yielded higher-performing classifiers than mean imputation across all five affective states. The results of the best classifier selected for each affective state are shown in Table 7.2, with Cohen's Kappa, Area Under Curve (AUC), accuracy, and F1 Scores shown. The approach to handling the multimodal data in this experiment was Early Fusion 1. The SVM achieved the highest classification performance for three affective states: *bored*, *confused*, and *surprised*. The J-Rip and J48 models achieved the highest performance for *frustrated* and *engaged concentration*, respectively.

Logistic regression performed relatively well for two affective states (*confused* and *bored)* but did not achieve the highest performance for any affective state. Notably, deep neural networks performed less effectively than the best classifier for each category and yielded poor results for a few affective states as well. This can possibly be attributed to an insufficient amount of training data, potentially introducing overfitting of the autoencoder or the neural network classifier.

Table 7.1. Comparison between mean imputation and autoencoder imputation for classifying student affective states.

Affective State	Mean Imputation	Autoencoder
Bored	0.087	**0.184**
Confused	0.068	**0.107**
Engaged Concentration	0.029	**0.037**
Frustrated	0.023	**0.049**
Surprised	0.019	**0.020**

Table 7.2. Results for best-performing classifier for each affective state using Early Fusion 1.

Affective State	Classifier	Kappa	AUC	Accuracy	F1 Score
Bored	SVM	0.134	0.621	0.639	0.368
Confused	SVM	0.134	0.621	0.639	0.368
Engaged Concentration	J48	0.146	0.565	0.579	0.601
Frustrated	J-Rip	0.078	0.555	0.917	0.138
Surprised	SVM	0.154	0.500	0.701	0.273

Following this procedure, we used each affective state's top-performing classifier to evaluate Early Fusion 2 and Late Fusion. Additionally, we evaluated Late Fusion based on two voting schematics: highest confidence (HC) and highest average confidence (HAC). Table 7.3 displays the results of Early Fusion 2 and both variations of Late Fusion. The results in Table 7.3 indicate that variations of data fusion do not improve the results of the classifier for any of the affective states, and in several cases, the results were significantly worse. One explanation for the relatively poor performance of Early Fusion 2 is that this method forces an even balance of attributes across modalities used to train the classifier. While PCA in Early Fusion 1 is able to select its own ratio of 30 principal components from the posture and EDA modalities to comprise

the 30 attributes for the classifier, Early Fusion 2 forces each PCA algorithm to select exactly 15 attributes per modality. Thus, if a modality such as the EDA data is inherently less informative than other modalities, Early Fusion 2 is replacing potentially useful attributes with less helpful attributes, resulting in lower performances across the classifiers.

Table 7.3. Comparison of multimodal data fusion techniques with best-performing classifiers for each affective state in terms of Cohen's Kappa.

	Bored	***Confused***	***Engaged Con.***	***Frustrated***	***Surprised***
Early Fusion 1	**0.1100**	**0.1340**	**0.1460**	**0.0780**	**0.1460**
Early Fusion 2	0.0650	0.0730	-0.020	0.0070	-0.020
Late Fusion (HC)	0.0960	0.02932	0.0651	0.0186	0.0651
Late Fusion (HAC)	0.1059	0.02932	0.0651	0.0259	0.0651

Previous work has found that EDA data does not have a tightly-coupled relationship with various affective states, as compared to other modalities such as facial expression (Harley et al., 2015). It is also a possibility that the EDA modality does not contain enough variance across multiple instances of each affective state for each classifier to distinguish between them effectively, as previously mentioned in Section 6.7. This problem is amplified during examples of mild or suppressed expressions of affective states. Additionally, modalities such as the Kinect posture data inherently contain higher dimensionality than the EDA data and therefore potentially contain more distinguishing factors between affective states.

Data fusion methods such as Early Fusion 2 and Late Fusion embrace an equal emphasis on all modalities present, which likely introduced a tradeoff between informative Kinect features and less informative EDA features that adversely impacted classifier performance for those two data fusion techniques. However, the EDA modality did appear to contain useful contextual

information that mostly improved classifier performance when used in conjunction with the Kinect posture modality.

To determine whether the addition of the EDA modality was indeed beneficial to the performance of each classifier, we trained a unimodal classifier on the complete posture data only and used the classifiers' performances as a baseline for each affective state. The baselines and best results from the multimodal approach for each affective state (Early Fusion 1) are shown in Table 7.4. The addition of the partially imputed EDA modality improved classifier performance on all affective states with the lone exception of *bored.* However, a significant majority of results indicate that multimodal data imputation for affect detection is beneficial relative to unimodal classification techniques.

Table 7.4. Comparison of Kinect-only unimodal vs. multimodal classifiers.

Bored **(SVM)**				
Classifier	**Kappa**	**AUC**	**Accuracy**	**F1 Score**
Unimodal	**0.1280**	0.6310	0.7817	0.2235
Multimodal	0.1100	0.6160	0.6897	0.2350
Confused **(SVM)**				
Classifier	**Kappa**	**AUC**	**Accuracy**	**F1 Score**
Unimodal	0.0280	0.5490	0.6151	0.1913
Multimodal	**0.1340**	0.6210	0.6398	0.3685
Engaged Concentration **(J48)**				
Classifier	**Kappa**	**AUC**	**Accuracy**	**F1 Score**
Unimodal	0.0480	0.5480	0.5496	0.6353
Multimodal	**0.0710**	0.5960	0.5774	0.6904
Frustrated **(J-Rip)**				
Classifier	**Kappa**	**AUC**	**Accuracy**	**F1 Score**
Unimodal	-0.001	0.4870	0.8783	0.0606
Multimodal	**0.0780**	0.5550	0.9174	0.0926
Surprised **(SVM)**				
Classifier	**Kappa**	**AUC**	**Accuracy**	**F1 Score**
Unimodal	-0.0210	0.4110	0.5913	0.0435
Multimodal	**0.1540**	0.5000	0.7007	0.2736

Prior research demonstrated the effectiveness of using the encoded latent feature vectors produced by an autoencoder to train a classifier (Jaques et al., 2017). We compare this approach to our approach of reconstructing the original dataset using decoding of latent representations. After producing a reconstructed dataset, we replace any values that have associated existing values in the original dataset. This process ensures that only the values determined to be missing or invalid are imputed, and values that existed in the original dataset are not overwritten with imputed values. Upon completion of this process, we train the same selected classifier model for each affective state on two variations of data: the encoded latent representations, and the decoded, reconstructed data. The comparison of each classifier's performance on the encoded and decoded data is shown in Table 7.5.

Table 7.5. Comparison of decoded vs. encoded dataset on classifier performance.

Bored **(SVM)**				
Classifier	**Kappa**	**AUC**	**Accuracy**	**F1 Score**
Decoded	**0.110**	0.616	0.689	0.235
Encoded	0.093	0.649	0.624	0.227
Confused **(SVM)**				
Classifier	**Kappa**	**AUC**	**Accuracy**	**F1 Score**
Decoded	**0.134**	0.621	0.639	0.368
Encoded	0.053	0.554	0.563	0.307
Engaged Concentration **(J48)**				
Classifier	**Kappa**	**AUC**	**Accuracy**	**F1 Score**
Decoded	**0.146**	0.565	0.579	0.601
Encoded	-0.020	0.492	0.533	0.658
Frustrated **(JRip)**				
Classifier	**Kappa**	**AUC**	**Accuracy**	**F1 Score**
Decoded	**0.078**	0.555	0.917	0.138
Encoded	-0.025	0.478	0.875	0.020
Surprised **(SVM)**				
Classifier	**Kappa**	**AUC**	**Accuracy**	**F1 Score**
Decoded	**0.154**	0.500	0.700	0.273
Encoded	0.007	0.394	0.667	0.054

The performance of the classifiers trained on the reconstructed dataset led the classifier to achieve higher performance for every affective state. A possible explanation includes the preservation of original values after the data reconstruction. This ensures that the dataset contains the original underlying, complex relationships between multiple attributes, which is often an important aspect of multimodal machine learning (Baltrušaitis et al., 2019). This problem extends to the encoded dataset, as reducing the dimensionality through the latent representation contains the inherent risk of losing contextual information that may affect the performance of a classifier.

7.4 Multimodal Datasets (*JavaTutor*)

For experiments with the *JavaTutor* dataset, we use facial expressions as the sensor-based modality, and two separate sensor-free modalities consisting of dialogue utterances and interaction trace data, respectively. Based on the prior work described in Chapter 5, we use BERT embeddings as the word embedding method for the dialogue modality. Based on the post-test surveys administered to the students in the study, we predict two individual affective states: *engagement* and *frustration*. For this work, we use a binary median split on *engagement* (M=3.79) and *frustration* (M=5.00) to convert the affect modeling to classification tasks, with the median split being determined by the training folds within each iteration of 10-fold cross-validation. To closer resemble the USMA dataset, and due to the fact that there is no actual missing data from the sensor-based *JavaTutor* modality, we apply artificial masking to the facial expression modality using a masking probability of 44%, the same percentage of missing data with the EDA modality.

7.5 Forward Feature Selection (*JavaTutor*)

Because of the large number of features in the multimodal data (19 facial expression features, 7 interaction features, 20 dialogue features), we implemented forward feature selection to eliminate

features with little or no predictive value and to reduce potential noise. Forward feature selection iterates through a list of features in a greedy manner, training a model on a single feature and continuing to add features if their inclusion increases the performance of the model on the target variable. This process continues until a predetermined number of features are selected or until all available features have been evaluated. This process has a few shortcomings. Due to the greedy nature of the algorithm, the features that are evaluated first have a higher chance of being selected. For example, the first feature that is evaluated is always retained, regardless of its true contribution to the predictive performance of the model. One approach to mitigating this issue is to perform forward feature selection for every possible combination of features, but this is often prohibitive as the number of combinations increases exponentially as the number of features increases, which imposes significant computational requirements. To mitigate the issue of bias in greedy feature selection while avoiding an exhaustive search across all feature combinations, we perform forward feature selection across a randomized ordering of all available features. We used a support vector machine (SVM) as the predictive model for each feature combination due to its effectiveness in high dimensional spaces and relatively small computational overhead. This process was repeated for 100 separate iterations and randomizations to ensure that each feature had an equal probability of being placed at a specific point within each feature ordering. Following this process, the features were sorted according to the frequency that each feature was selected across all 100 iterations (Henderson et al., 2021a). To compensate for the difference in the number of features for each data channel, we performed forward feature selection on the facial expression and dialogue modalities and selected the ten most frequently selected features. It should be noted that because we selected the ten most frequent features from the facial expression modality and the dialogue modality, and the interaction-based modality contained only 7 total features, each feature from the latter modality

was included in the data modeling process Because certain features such as AU durations and coding times increase monotonically throughout a learning trajectory and can lead to indirect data leakage, the features were scaled by the total elapsed time up to the current timestamp, so these features were converted to proportional representations of the elapsed time at each time interval. This feature selection process took place within each cross-validation fold, and as a result, each fold produced a different combination of selected features.

7.6 Model Evaluation (*JavaTutor*)

Prior to the affect model training, denoising autoencoders were trained using the same procedure outlined in Section 7.1 and Section 7.2 following the artificial masking of the facial expression modality as previously described. The autoencoder models consisted of hidden layers of 32, 16, and 8 nodes, Leaky ReLU activation functions, and were trained using a learning rate of 0.1. Following the imputation process, the reconstructed dataset underwent the preprocessing features described in the prior sections and was used to train the affect detection models.

The affect detection models were evaluated using 10-fold nested cross-validation, with the splits for each fold occurring at the student level to ensure that a student's learning sessions were contained only within a single training, validation, or test set. The dataset was standardized within each cross-validation fold by subtracting the mean from each feature and dividing by the feature's standard deviation, as determined by the training data. This rescales the data to have a standard deviation of 1 (unit variance) while centering the mean to be 0. The standardized training data is then used for forward feature selection as described in Section 7.5.

After feature selection, a classifier model was trained on the multimodal data. We evaluated five different models: SVM, logistic regression, Naïve Bayes, random forest, and feedforward neural network. We performed hyperparameter tuning using a 3-fold nested cross-validation

within the training set for each outer cross-validation fold. The hyperparameters that were varied for each model included the regularization parameter and kernel (SVM), regularization parameter (logistic regression), number of estimators (random forest), and number of layers and nodes (feedforward neural network), and were evaluated using a iterative grid search based on the performance on the validation data folds for each outer cross-validation split. Early stopping was implemented for the FFNN models based on a patience of 10 epochs using the validation performance to prevent overfitting during the training process. The same data fusion techniques mentioned in Section 6.5 are evaluated with the *JavaTutor* dataset as well. All outer and inner cross-validation folds are retained across all experiments to ensure consistency.

7.7 Results (*JavaTutor*)

For these experiments, three different baseline configurations are evaluated using the same 5 models for each configuration. The first is a unimodal baseline using the dialogue and interaction-based modalities (it is assumed that the dialogue and interaction-based modalities can be combined as they both originate from the same data source). The second baseline is based on training models using only complete multimodal data (i.e., removing all artificially-masked data points). The third baseline is mean imputation, a naïve approach to imputation as used in Section 7.3. The results of these experiments are shown in Table 7.6 below. Due to the use of a median split and the subsequent absence of significant imbalances in the data, we focus on F1 Score and raw accuracy as the primary evaluation metrics for this work.

Similar to the experiments outline in Chapter 6 and Section 7.3, we note that the autoencoder-based data imputation leads to higher performance for both *engagement* and *frustration* across all baselines. The ratio of missing facial expression data appears to have a significant impact on the performance of the affect models, as the complete data baseline achieves

Table 7.6. Experimental results of autoencoder-based imputation on *JavaTutor* dataset.

Engagement						
Imputation Method	**Model**	**Data Fusion**	**F1 Score**	**Accuracy**	**AUC**	**Kappa**
Unimodal	SVM	N/A	0.645	0.552	0.532	0.046
Complete Data	RF	EF1	0.548	0.508	0.543	0.043
Mean Imputation	FFNN	EF1	0.657	0.574	0.491	0.027
Autoencoder Imputation	**FFNN**	**LF**	**0.699**	**0.619**	**0.562**	**0.026**
Frustration						
Imputation Method	**Model**	**Data Fusion**	**F1 Score**	**Accuracy**	**AUC**	**Kappa**
Unimodal	FFNN	N/A	0.602	0.526	0.534	0.019
Complete Data	FFNN	EF1	0.443	0.506	0.568	0.109
Mean Imputation	FFNN	EF1	0.604	0.506	0.54	-0.007
Autoencoder Imputation	**FFNN**	**EF1**	**0.706**	**0.612**	**0.624**	**0.171**

the worst performance by a significant margin in both evaluations. Additionally, it is notable that the unimodal and the mean imputation baselines achieve similar performance, indicating that the facial expression modality possibly contains a significant amount of variance that is eliminated or underestimated by the mean imputation process, and therefore eliminates significant predictive patterns that are only captured by the autoencoder imputation model.

It is also notable that feedforward neural networks were the optimal model for each of the *frustration*-based experiments, and for half of the *engagement*-based experiments. This is in stark contrast to the USMA data results from Section 7.3, where the FFNN was not selected as the optimal affect model in any of the evaluations. This is likely attributed to the fact that the *JavaTutor* data contains roughly twice the available data as the USMA data, which is critical for the training of more complex learning methods such as deep learning models. Similarly to the results in Section 7.3, Early Fusion 1 was often selected as the optimal multimodal data fusion method for these experiments, with the line exception of the autoencoder-based imputation for *engagement*. Based on the results from Sections 7.3 and this section, we are able to verify the generalizability of the deep learning approach to multimodal data imputation across two diverse datasets and populations.

CHAPTER 8

ENHANCING MULTIMODAL AFFECT DETECTION: SENSOR-BASED AND SENSOR-FREE MODALITIES

Although the multimodal affect detection models demonstrated improved performance when integrating the EDA modality into the input USMA data, overall performance improvements were not significant. This could be attributed to a number of factors, including the singular dimensionality of the EDA data channel, the amount of data samples that contained missing or invalid EDA data, the level of noise in the captured EDA readings, and the potential decrease in predictive performance that could be attributed to the affect models' training on imputed data.

To further investigate multimodal affective prediction, we removed the EDA modality and replaced it with a sensor-free, interaction-based modality. We hypothesized that this would lead to improved predictive performance due to the lack of noise in the data compared to the EDA modality, the increased number of features available from the interaction trace data captured during gameplay, and the availability of interaction data for each student. In this section, we perform an investigation into multimodal affect detection similar to that described in Chapter 6, and we investigate the effectiveness of a temporal-based "velocity" modality that was generated in a post-hoc manner (Henderson, Rowe, et al., 2019). Additionally, we perform a similar investigation using the *JavaTutor* datasets, while retaining the facial expression modality and evaluating the predictive value of the dialogue and interaction-based modality as well.

8.1 Model Evaluation (USMA)

The features used in this work are described in Sections 4.1.5.1, 4.1.5.2, and 4.1.5.4. We introduce the temporal-based features to the data because of their ability to explicitly quantify gesture and

action movement exhibited by the students during their gameplay sessions. Because of the imbalance between positive and negative instances of several affective states, including *frustrated* and *surprised*, each dataset underwent upsampling using the Synthetic Minority Oversampling Technique (SMOTE) (Chawla et al., 2002), within training sets only. This process selects a positive instance of the minority class at random and linearly interpolates synthetic data points between the selected point and another minority sample chosen by a randomized K-nearest neighbor clustering approach. SMOTE is a common approach to resolving class imbalance issues by bringing the class distribution to a uniform balance while avoiding duplication of minority instances, which can lead to overfitting in affective models.

The datasets for each binary classification task were split into a training set and a held-out test set, containing approximately 80% and 20% of the total dataset, respectively. The datasets were sampled to ensure that the distributions between training and test data were relatively similar. The training set was used to evaluate each of the modeling approaches using four-fold cross-validation. The splits for both the cross-validation and training/test sets were performed at a student level to avoid data leakage from a single session during either the training or evaluation phases.

Prior to training, each of the datasets was normalized and underwent forward feature selection to allow the models to train using only selected features, eliminating redundant or otherwise uninformative features. Additional detail about this process can be found in Section 6.4. We used an SVM to guide feature selection due to its ability to efficiently perform non-linear classification. For this work, the forward feature selection was implemented using Python 3.6 instead of RapidMiner 9.0 to provide increased flexibility for various aspects of the data processing pipeline. We selected 12 features per data channel. In cases involving multimodal input, 6 features were selected per feature type (i.e., posture and temporal) for the posture-based feature

representations, and 4 features were selected per feature type across both of the data channels (i.e., posture, temporal, and interaction). This approach is selected due to prior work that demonstrated the effectiveness of treating the temporal features as a separate "modality" (Henderson, Rowe, et al., 2019). Feature normalization, SMOTE upsampling, feature selection, and model training took place within each cross-validation fold to prevent data leakage across the training and validation data.

We investigated several approaches for integrating feature representations from the independent modalities using multimodal data fusion techniques. Additional detail on the implementation of the multimodal data fusion can be found in Section 5.1.3, with the only significant difference being the use of the interaction-based modality instead of the EDA-based modality. We used the highest average confidence across each class to determine the final representative prediction within Late Fusion.

We compared five machine learning techniques for inducing detectors of each affective state: SVM, random forest (RF), Gaussian Naïve Bayes (NB), logistic regression (LR), and FFNN. To serve as baselines, we trained unimodal models using either interaction data or posture data. These models were based on unimodal affect detectors induced in prior work, although we made several methodological refinements related to feature selection, upsampling, cross-validation, evaluation on a held-out test, and implementation of machine learning models. These modifications have a small impact on the results for the baseline models, but overall accuracy trends across affective states remained the same as in prior findings. The posture-only baselines were evaluated using both spatial and temporal modalities using data fusion techniques depicted in Section 6.5, but for this analysis, we considered these models to be "unimodal" because both the spatial and temporal features were extracted from the same sensor-based data channel. Each

model's predictive accuracy was examined under cross-validation on the training set to determine which model was "optimal" for the respective combination of feature set, data fusion method, and affective state. The model with highest performance during cross validation was evaluated using data from the held-out test set.

8.2 Results (USMA)

Each model was evaluated with Cohen's Kappa as the primary metric, due to its ability to account for positive classifications occurring due to random chance or dataset-induced bias. We also present results in terms of raw classification accuracy and F1 Score. Results from this evaluation are shown in Table 8.1, with the highest-performing combination of data fusion technique and model for each affective state shown in bold.

We observe from the results that multimodal affect detectors utilizing a combination of interaction-based and posture-based modalities outperformed posture-only baseline and interaction-only baseline models for four out of the five affective states, with the sole exception being the state of *confused*. For the four other affective states, Early Fusion 1 was the best fusion technique for *surprised*, and Early Fusion 2 was the most accurate method for *bored* and *frustrated.* Late Fusion achieved the highest performance for *engaged concentration*. The majority of the affective states produced a relatively high Kappa value (> 0.2), excluding *surprised*.

It is noteworthy that the FFNN models were the optimal classification model for a majority of cases (60%), potentially due to their ability to robustly model complex, non-linear relationships between modalities. This capability is especially important when modeling data from multiple independent modalities such as Early Fusion and the posture-based models using both spatial and temporal modalities. SVM and RF models were occasionally the best-performing classification

Table 8.1. Optimal models for each combination of modalities and affective states (USMA).

Bored

Modality	Model	Kappa	Accuracy	F1 Score
Gameplay	RF	0.378	0.857	0.447
Posture (EF1)	FFNN	0.314	0.907	0.347
Posture (EF2)	SVM	0.294	0.901	0.333
Posture (LF)	FFNN	0.110	0.845	0.193
Multimodal (EF1)	LR	0.432	0.858	0.510
Multimodal (EF2)	**SVM**	**0.466**	**0.907**	**0.516**
Multimodal (LF)	FFNN	0.456	0.913	0.500

Confused

Modality	Model	Kappa	Accuracy	F1 Score
Gameplay	FFNN	0.016	0.458	0.323
Posture (EF1)	SVM	0.133	0.697	0.328
Posture (EF2)	**FFNN**	**0.220**	**0.759**	**0.360**
Posture (LF)	FFNN	0.114	0.753	0.230
Multimodal (EF1)	FFNN	0.118	0.709	0.298
Multimodal (EF2)	FFNN	0.102	0.500	0.400
Multimodal (LF)	FFNN	0.132	0.543	0.412

Engaged Concentration

Modality	Model	Kappa	Accuracy	F1 Score
Gameplay	RF	0.104	0.571	0.604
Posture (EF1)	FFNN	0.156	0.567	0.520
Posture (EF2)	SVM	0.156	0.586	0.641
Posture (LF)	FFNN	0.119	0.574	0.653
Multimodal (EF1)	LR	0.119	0.574	0.653
Multimodal (EF2)	**SVM**	0.162	0.604	0.711
Multimodal (LF)	FFNN	**0.254**	**0.617**	**0.569**

Frustrated

Modality	Model	Kappa	Accuracy	F1 Score
Gameplay	RF	0.051	0.664	0.118
Posture (EF1)	FFNN	0.149	0.928	0.166
Posture (EF2)	SVM	0.082	0.913	0.125
Posture (LF)	FFNN	0.082	0.913	0.125
Multimodal (EF1)	LR	0.112	0.709	0.203
Multimodal (EF2)	**SVM**	**0.205**	**0.895**	**0.260**
Multimodal (LF)	FFNN	0.002	0.339	0.115

Surprised

Modality	Model	Kappa	Accuracy	F1 Score
Gameplay	RF	0.079	0.836	0.135
Posture (EF1)	FFNN	0.083	0.660	0.153
Posture (EF2)	SVM	0.023	0.864	0.083
Posture (LF)	FFNN	0.005	0.098	0.087
Multimodal (EF1)	**LR**	**0.104**	**0.925**	**0.142**
Multimodal (EF2)	SVM	-0.037	0.925	0.000
Multimodal (LF)	FFNN	0.080	0.913	0.125

techniques for both unimodal and multimodal affect detection. NB and LR models were each selected once as the best-performing model for a certain multimodal configuration, although neither model was the optimally performing model for an entire affective state.

To conduct a more in-depth analysis of the predictive value of each modality during multimodal data fusion, we recorded the frequency that each feature was selected during cross-validation for each data fusion variation. Although Early Fusion 2 and Late Fusion enforced an inherent balance between modality features, Early Fusion 1 combined all features into a single dataset prior to feature selection, resulting in a majority of features being weighted towards the most predictive modality.

We find that the ratio of interaction-based features to posture-based features selected for all 4 folds (48 total features) is 25:23 for *bored*, 18:30 for *confused*, 22:26 for *engaged concentration*, 26:22 for *frustrated*, and 27:21 for *surprised*. The distribution of features was skewed towards interaction-based features for three of the affective states and toward posture-based features for two of the affective states, suggesting a comparable degree of predictive value between modalities across all affective states. This trend may explain why Early Fusion 2 and Late Fusion yielded the best-performing models for three of the five affective states examined (i.e., *bored*, *engaged concentration*, and *frustrated*).

Results indicate that *confusion* was modeled most effectively using posture features only, which suggests that student posture may be more indicative of confusion than interaction-based features extracted from *TC3Sim* log data. D'Mello and Graesser (2010) previously demonstrated a correlation between students' upright posture and instances of displayed confusion. In aggregate, the results indicate that the predictive value of each modality varies across affective states, which in turn impacts the performance of Early Fusion and Late Fusion techniques. Utilizing dedicated

models for each affective state, rather than inducing a single model to classify all affective states, enables the use of different modeling and data fusion techniques to yield improved detector performance.

It was observed that the most frequently selected features across all of the affective states were *sitmid_freq* (frequency a student sat upright), *sit_forward_freq* (frequency a student leaned forward), *Sum of isSafe* (number of times a student's in-game character was safe), *CENTER_SHOULDER_max* (maximum distance of the *CENTER_SHOULDER* vertex from the Kinect sensor), *sitmid_freq_20sec* (frequency a student sat upright within a 20-second window), and *Min of HeartRate* (minimum heart rate of an NPC player receiving medical care). This indicates that each modality contributed relatively equally to the performance of the optimal multimodal classifiers. The two most frequent features (*sitmid_freq, sit_forward_freq*) were representative of the frequency that a student adjusted their posture, while the two interaction-based features (*Sum of isSafe, Min of HeartRate*) were representative of the student's in-game actions and states, respectively. The remaining two features were also posture-based features: *CENTER_SHOULDER_max* focused on the furthest distance of the *CENTER_SHOULDER* vertex from the Kinect sensor over the entire session, and *sitmid_freq_20sec* focused on the students' frequency of sitting upright for the preceding 20 seconds. A possible explanation for the improvement of the multimodal models' performance over the unimodal baselines is that the multimodal models were able to obtain a more thorough, comprehensive picture of the students' behavior, as the most frequently used features were widely varied in the information provided.

8.3 Model Evaluation (*JavaTutor*)

For the evaluations using the *JavaTutor* data, the processing pipeline for this dataset is similar to the one described in Section 7.1 and 7.2. Additionally, the models evaluated are the same as in

Section 7.8, using the same nested 10-fold cross-validation splits. For this work, the interaction-based modality and the dialogue-based modality are treated independently, and the three data fusion methods discussed in Section 6.5 are evaluated for both bimodal (dialogue + interaction) and trimodal (dialogue + interaction + facial expression) configurations.

8.4 Results (*JavaTutor*)

The results for the unimodal baselines and multimodal evaluations are shown in Table 8.2. The dialogue-only (D) and interaction-only (I) modalities are used as unimodal baselines; facial expressions (F) are not considered as a unimodal baseline as sensor-based modalities are not typically utilized individually without the presence of a sensor-free modality such as interaction trace data.

Table 8.2. Optimal models for each combination of modalities and affective states (*JavaTutor*).

Engagement

Modalities	Model	Data Fusion	F1 Score	AUC	Accuracy
Interaction-Only	FFNN	N/A	0.656	0.416	0.568
Dialogue-Only	SVM	N/A	0.642	0.523	0.537
I + D	SVM	LF	0.684	0.525	0.559
I + D + F	**RF**	**LF**	**0.687**	**0.531**	**0.564**

Frustration

Modalities	Model	Data Fusion	F1 Score	AUC	Accuracy
Interaction-Only	SVM	N/A	0.639	0.575	0.556
Dialogue-Only	SVM	N/A	0.650	0.571	0.540
I + D	SVM	EF2	0.699	0.594	0.590
I + D + F	**RF**	**EF1**	**0.709**	**0.589**	**0.596**

F1 Score is selected as the primary evaluation metric due to the median split that is used for each affect state, effectively resolving any significant issues with class imbalances in this case. For both affective states, the multimodal approaches, both bimodal and trimodal, outperform all unimodal baselines. It is noticeable that the facial expression does induce higher predictive

performance across both affective states, the improvement appears to be marginal in both cases. Additionally, the SVM model was selected as the most predictive model for a vast majority of the evaluations, which is a stark contrast to the same evaluations used with the autoencoder-imputed multimodal data described in the previous chapter. This likely can be attributed to the overall predictive value of the imputed values compared to the ground truth, in addition to the significant amount of data that is imputed (44%).

For *engagement*, the optimal multimodal data fusion method was Late Fusion for both multimodal evaluations, which indicates that modeling each individual modality separately prior to generating a single representative prediction for the entire framework induces higher performance. This may indicate a lack of inter-modal relationships between the individual modalities, such that a student's facial expressions when engaged with the *JavaTutor* platform is not saliently correlated with certain dialogue patterns or user interaction. Conversely, Early Fusion induced the highest performance from the affect models of *frustration*, indicating the presence of inter-modal patterns across the multimodal data that are indicative of the presence of *frustration*. This also points to the value of capturing modalities other than the interaction-based modality during student modeling tasks, particularly in cases where certain features may be infrequent or contain high variance. One example of this is the programming-based features in the *JavaTutor* corpus. For example, the number of compile attempts or the number of successful runs may not be particularly predictive within a single modality, as each of these events tends to occur relatively infrequently throughout a single learning session. However, the predictive value of these features appears to be reinforced through combination with other modalities. As a result, we are able to conclude that a multimodal approach to student affect detection leads to higher predictive performance, across both the USMA and *JavaTutor* datasets.

CHAPTER 9

GENERATIVE DATA AUGMENTATION FOR MULTIMODAL AFFECT MODELING

While the use of interaction-based and posture-based multimodal data appears to improve the performance of the affect detection models, we seek to further enhance their predictive performance through the use of deep generative models. Generative modeling techniques such as GANs or VAEs show particular promise within this particular predictive task due to their ability to produce synthetic data that retains predictive traits within complex, non-linear data.

The effectiveness of generative models within the multimodal affect detection task is evaluated in two separate phases: 1) data augmentation, in which generative models are used to increase the size of the multimodal training datasets, and 2) data imputation, in which generative models are used to impute artificially corrupted data within a particular modality. In this manner, deep learning is used as an external means of enhancing multimodal affect detection models, instead of simply serving as an alternative modeling approach. Because of the promise that interaction-based and posture-based multimodal models demonstrated in prior work (Chapter 8), we continue to utilize these two modalities in the following sections.

9.1 Data Augmentation for Multimodal Affect Detection

Multimodal machine learning approaches to affect detection require large quantities of training data from each modality. However, multimodal sensor data is often compromised because of several problems, including calibration issues, sensor noise, missing or imbalanced data, and data storage constraints, as demonstrated in Chapter 4. Similarly, interaction-based modalities are often affected by issues such as hardware failure, software errors, and logging issues. These challenges can significantly impact the amount of or quality of data available to train multimodal affect

detection models, raising concerns about overfitting and data sparsity that can adversely impact the accuracy and robustness of student affect models.

We address the issue of insufficient data for multimodal affect detection with Auxiliary Classifier Generative Adversarial Networks (AC-GANs). AC-GANs, which can effectively model multimodal data distributions, are utilized to generate synthetic data consisting of posture and interaction data captured from students engaging with *TC3Sim* and interaction, dialogue, and facial expression data captured from students engaging with *JavaTutor*. To ensure the quality of the augmented dataset produced by the AC-GAN, we demonstrate the effectiveness of a filtering method based on the Wasserstein distance metric to ensure that the augmented multimodal data follows the original data sample's distribution (S. Vallender, 1974). Finally, we demonstrate the effectiveness of using the AC-GAN discriminator network as a classification model for detecting students' run-time affective states during interactions with *TC3Sim* and *JavaTutor*.

9.2 Auxiliary Classifier Generative Adversarial Networks

AC-GANS (Odena, Olah, & Shlens, 2017) are an extension of generative adversarial networks which consist of two deep neural network models, a generator and a discriminator, that compete against one other in an adversarial fashion within a zero-sum setting to generate synthetic data that resembles the original data used for training (Goodfellow et al., 2014). Section 2.3.4 contains a more in-depth explanation of standard GAN models. Conditional GANs extend the GAN architecture by providing associated information to both the generator and discriminator, such as a class label associated with the desired synthetic output (Mirza & Osindero, 2014). AC-GANs deviate from the conditional GAN architecture by allowing the discriminator to predict the class label of the generated sample as well as the data source (i.e., "real" or "fake" status). The generator aims to minimize the ability of the discriminator to distinguish between real and fake data, while

maximizing its ability to predict the class label of the generated data. This often leads to a more stabilized training process and also allows the GAN to learn a latent space representation that does not rely on the class label as input, unlike a standard conditional GAN. The discriminator's ability to be trained to predict the class label of the generated data lends itself for additional use as an affect classification model, a property that is investigated within this work.

9.3 Wasserstein Filtering

To ensure that our synthetic data accurately reflected the distribution of the original dataset, we utilized a filtering process based on the Wasserstein metric (S. Vallender, 1974). Also known as the "earth mover's distance," this metric is grounded in optimal transport theory and is a method to quantify the distance between two continuous probability distributions. We selected this metric due to its ability to account for both the probability density of the synthetic data compared to the original distribution and also the distance within a defined metric space, giving it an advantage over related methods such as Kullback-Leibler divergence. Once the AC-GAN was used to generate batches of 50 augmented data samples, using a Gaussian noise vector and the minority class label as the conditioning variable to the generator, the average Wasserstein distance between each feature and the corresponding feature in the original dataset was computed across all features for each generated sample in a single batch. After this process was repeated for 10 batches, the batch with the lowest average Wasserstein distance is selected for inclusion in the augmented dataset. This process continues iteratively until all classes in the dataset were uniformly distributed. This method ensures that the synthetic data contains an appropriate variance level beneficial for the affective models while not creating closely identical examples of the original minority data, which could induce overfitting in the affective models.

9.4 Model Evaluation (USMA)

Using the now-balanced datasets, we evaluated several different machine learning models and determine which model produces binary affect classifiers with the greatest accuracy. We used five different models for each affective state (e.g., *bored, confused, engaged concentration, frustrated, surprised*): support vector machine, random forest, Gaussian Naïve Bayes, logistic regression, and feedforward neural network. For evaluations involving an AC-GAN model, the trained discriminator was retained from the data augmentation phase and trained further using the same data as the other five models. The discriminator's predictions of the class label were used during the affect model evaluations. The models' classification performance was measured in terms of AUC as the primary evaluation metric to account for model correctness in the face of class imbalance. We also included the predictive accuracy as well as the F1 score, recall, and precision for each model to illustrate the tradeoffs inherent among different evaluation metrics. To serve as a baseline against which we compared our affective models' classification performance, we trained the same set of models on the raw normalized dataset without any prior data augmentation. We compare our data augmentation framework against two common approaches for resolving class imbalance issues in affective modeling: minority cloning and Synthetic Minority Over-Sampling Technique, which are the upsampling techniques used in the prior work shown in Chapter 8. The models were trained using 5-fold cross-validation with data splits maintained on a student level to prevent data leakage among individual gameplay sessions. 5-fold cross validation was chosen to maintain an adequate number of positive instances of each affective state within each validation fold. The class distribution within each fold was maintained using a stratified sampling approach. Prior to training, the dataset used to induce each affect detection model was normalized so that each feature's range fell between [-1, 1]. Following normalization, feature selection was performed

by retaining a subset of features that contained the highest chi-squared test values with the class variable. Normalization, feature selection, and class balancing took place with the training set of each cross-validation step to ensure that data leakage did not occur during baseline or AC-GAN data augmentation. The data augmentation process using the AC-GAN is illustrated in Figure 9.1.

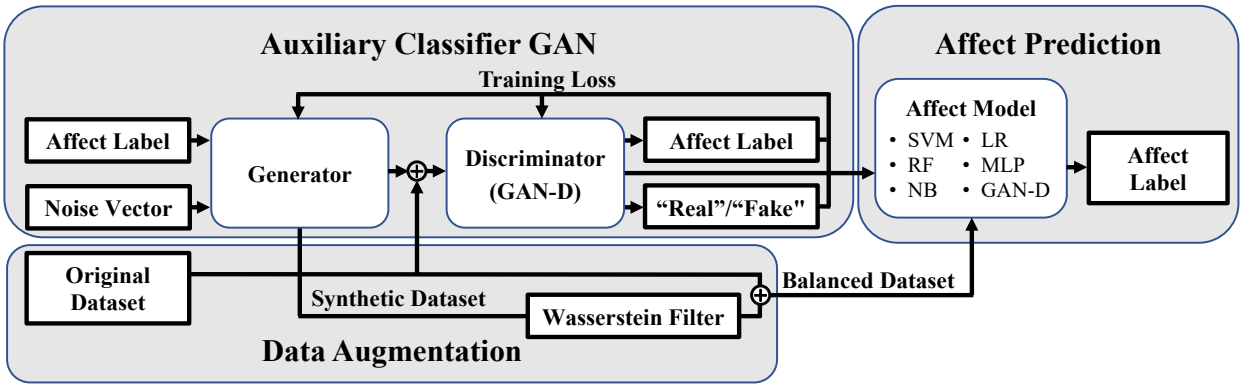

Figure 9.1. Data augmentation process for affect detection model training with an AC-GAN.

9.5 Results (USMA)

For each upsampling technique and affective state, we evaluate five classification techniques (SVM, RF, NB, LR, and FFNN) in addition to the AC-GAN discriminator. The highest-performing models are presented in Table 9.1, with the results of the optimal classifiers in terms of area under the curve shown in bold. We compare our results to the baseline classifiers trained on the original dataset as well as classifiers trained on augmented data that underwent different forms of upsampling. AC-GAN data augmentation outperformed each of the other upsampling approaches in terms of AUC with the exception of *bored*. For each of the 4 remaining affective states, the AC-GAN combined with the Wasserstein filtering (AC-GAN-W) outperformed the standard AC-GAN in 3 of 4 cases. Additionally, the AC-GAN discriminator was selected as the optimal affect model in 5 of 10 possible cases across the two AC-GAN upsampling tests for the 5 distinct affective states.

Table 9.1. Results of upsampling techniques for each affective state (USMA).

Bored				
Upsampling	**Model**	**AUC**	**Accuracy**	**F1 Score**
Baseline	NB	0.701	0.542	0.277
Cloning	SVM	0.815	0.827	0.461
SMOTE	**LR**	**0.817**	**0.839**	**0.474**
AC-GAN	NB	0.774	0.831	0.421
AC-GAN-W	GAN-D	0.662	0.802	0.330
Confused				
Upsampling	**Model**	**AUC**	**Accuracy**	**F1 Score**
Baseline	NB	0.520	0.509	0.229
Cloning	NB	0.518	0.507	0.229
SMOTE	NB	0.518	0.513	0.228
AC-GAN	FFNN	0.533	0.662	0.232
AC-GAN-W	**GAN-D**	**0.543**	**0.588**	**0.326**
Engaged Concentration				
Upsampling	**Model**	**AUC**	**Accuracy**	**F1 Score**
Baseline	SVM	0.569	0.610	0.705
Cloning	SVM	0.569	0.610	0.705
SMOTE	SVM	0.554	0.578	0.650
AC-GAN	**GAN-D**	**0.575**	**0.616**	**0.720**
AC-GAN-W	GAN-D	0.571	0.523	0.461
Frustrated				
Upsampling	**Model**	**AUC**	**Accuracy**	**F1 Score**
Baseline	FFNN	0.573	0.920	0.132
Cloning	LR	0.614	0.717	0.119
SMOTE	LR	0.654	0.847	0.194
AC-GAN	GAN-D	0.699	0.818	0.183
AC-GAN-W	FFNN	**0.748**	**0.664**	**0.173**
Surprised				
Upsampling	**Model**	**AUC**	**Accuracy**	**F1 Score**
Baseline	NB	0.517	0.389	0.077
Cloning	NB	0.530	0.388	0.080
SMOTE	SVM	0.501	0.801	0.054
AC-GAN	NB	0.562	0.679	0.077
AC-GAN-W	FFNN	**0.617**	**0.578**	**0.105**

An AC-GAN discriminator was not trained for the baseline, cloning, or SMOTE upsampling experiments. This 50% selection rate was the highest selection rate among all affective models, compared to 20% (5/25) for SVM, 32% (8/25) for Naïve Bayes, 12% (3/25) for logistic regression, and 16% (4/25) for feedforward neural network.

The results indicate that AC-GANS are the highest-performing upsampling approach for 4 of the 5 affective states. The lone exception, *bored*, demonstrated very high AUC values for all of the upsampling techniques. In this case, the AC-GAN upsampling technique outperformed the baseline, but the other upsampling techniques induced higher AUC scores from the models than the AC-GANs. This may be due to predictive anomalies in the data or boredom-specific behavioral cues, which warrant further investigation.

The impact of the AC-GAN augmentation was more significant for *frustrated* and *surprised*, the two most imbalanced classes in the dataset. One explanation for this behavior is that the data points belonging to the minority class are likely highly localized, as these instances of *frustrated* and *surprised* comprise 4.2% and 3.8% of the total dataset, respectively. Because SMOTE is based on a K-nearest neighbor approach, the augmented data will be contained within the same range as the original data points.

Minority cloning does not introduce variance during data augmentation, and as a consequence, may cause classifiers to overfit minority data, which is likely to harm affect detector accuracy. While SMOTE introduces some variance during data synthesis, it is limited by its dependence upon producing samples using linear interpolation. This could also lead to the predictive model conforming to the localization of the minority class and overfitting of the model. One aspect of using generative models such as AC-GANs for data augmentation is their capacity to model complex relationships between various data attributes through non-linear transformations and generate synthetic data according to the underlying distributions while still maintaining a beneficial amount of variance for the classifiers. It should be noted that the results indicate that the Wasserstein distance-filtering approach is an effective method of enforcing an adequate variance level in the synthetic data while still retaining accurate modeling of the original data distributions.

This allows the AC-GAN-based data augmentation process to be more robust when encountering heavily skewed data.

Of note is the performance of the AC-GAN discriminator as the most frequently selected optimal affect model. Using subsets of real and artificial data from the generator, the weights of the AC-GAN are initially trained within a multi- task framing, meaning that the discriminator learns to not only distinguish trends of the real and artificial data, but also the binary class label of each sample. This factor may play a role in the enhanced performance of the AC-GAN discriminator as an affect model.

9.6 Model Evaluation (*JavaTutor*)

For the model evaluations using the *JavaTutor* data, the processing pipeline for this dataset is similar to the one described in Section 7.5 and 7.6. The models evaluated are the same as in Section 7.6, using the same nested 10-fold cross-validation splits. A primary difference in this work compared to the prior *JavaTutor* experiments is that we do not use a median split to convert the target variables to binary values; this is due to the fact that a median split does not result in realistic data imbalances like those of the USMA data. Because this work is designed to address data imbalances in particular, we use a thresholding approach to split the survey responses into "low" and "high" categories using a cutoff of 4 for *engagement* and 50 for *frustration*. This results in a distribution of 40.1% positive instances of *engagement* and 8.48% positive instances of *frustration*, which are similar to the distributions in the USMA dataset. The AC-GAN implementation and the Wasserstein filtering approach are mostly unchanged from the USMA dataset evaluations, in addition to the use of the trained discriminator component of the AC-GAN as an alternative affect detection model. The baseline upsampling approaches (SMOTE, minority cloning) are included in the *JavaTutor* evaluations as well.

9.7 Results (*JavaTutor*)

The results of the *JavaTutor* evaluations of the AC-GAN approach in addition to the two upsampling baselines are shown in Table 9.2. The AC-GAN approach combined with the Wasserstein filtering achieved the highest performance for both affective states in terms of AUC. The discriminator component of the AC-GAN was the optimal affect model for *engagement*, but did not outperform the FFNN model for *frustration*, and was not the optimal model for either of the evaluations using the AC-GAN model without filtering. The discriminator component was selected for only 25% of the total possible evaluations, compared to the 50% selection rate for the USMA data, but the number of evaluations should also be taken into account, as the USMA dataset contains 60% more affective states and experiments compared to the *JavaTutor* dataset. With this in mind, it is notable that the next most frequent models (Naïve Bayes and SVM) were only selected for 20% out of all possible evaluations.

Similarly to the USMA results described in Section 9.5, the data augmentation using the AC-GAN approach achieved higher performance on the more imbalanced affective state

Table 9.2. Results of upsampling techniques for each affective state (*JavaTutor*).

Engagement				
Upsampling	**Model**	**AUC**	**Accuracy**	**F1 Score**
Baseline	FFNN	0.500	0.459	0.554
Cloning	NB	0.529	0.469	0.549
SMOTE	NB	0.542	0.467	0.537
AC-GAN	RF	0.530	0.538	0.538
AC-GAN-W	**GAN-D**	**0.545**	**0.536**	**0.536**
Frustration				
Upsampling	**Model**	**AUC**	**Accuracy**	**F1 Score**
Baseline	LR	0.669	0.907	0.117
Cloning	SVM	0.671	0.884	0.163
SMOTE	SVM	0.649	0.904	0.083
AC-GAN	RF	0.675	0.879	0.165
AC-GAN-W	**FFNN**	**0.692**	**0.867**	**0.190**

(*frustration*) compared to the more balanced affective state (*engagement*). This may reinforce the possibility of naïve upsampling approaches such as SMOTE or minority cloning becoming concentrated on the localized patterns within the training data, a problem that is hopefully mitigated by the use of more complex modeling approaches and the variance levels enforced by the Wasserstein filtering method. Ultimately, we are able to see that the AC-GAN approach combined with Wasserstein filtering generalizes well across both data corpora, and the use of the discriminator serves as a potential alternative for affect detection tasks.

CHAPTER 10

MULTIMODAL DATA IMPUTATION WITH CONDITIONAL GENERATIVE MODELING

In addition to data augmentation, deep generative models show significant promise in other domains, such as missing data imputation. We investigate multimodal data imputation with deep conditional generative adversarial networks to address potential data loss issues in the multimodal affect detection task. We utilize these deep conditional GANs because of their capability to effectively model complex relationships between multiple input variables and data channels. The application of conditional generative models to multimodal data streams has received comparatively little attention, particularly in the context of modeling student affect in adaptive learning environments.

Specifically, we investigate two types of conditional generative models for multimodal data imputation: conditional GANs (C-GANs) and conditional VAEs (C-VAEs). The models are evaluated within our multimodal affect detection framework that tracks posture data and interaction data from students engaged with the *TC3Sim* game-based learning environment, in addition to the interaction, dialogue, and facial expression data from the *JavaTutor* learning platform. Similar to prior work, the USMA affect detection models are induced to predict learning-centered affective states obtained from the BROMP field observations of each student, while the *JavaTutor* affect detection models predict the self-reported affect states from the post-test surveys. The models are evaluated using varying levels of "missingness" to demonstrate the impact that intact data availability has on each generative model. The effectiveness of each imputation method is evaluated based on its impact on the predictive performance of multimodal affect detectors that are previously trained on all available multimodal data without masking. Results indicate that the

non-linear generative models based on deep neural networks show significant promise compared to several competing linear baseline approaches for data imputation.

10.1 Conditional Generative Adversarial Networks

Conditional GANs expand upon the traditional GAN model by training the discriminator on additional data, or "conditions", associated with the input feature vector, such as a class label (Mirza & Osindero, 2014). Likewise, the generator is trained on the same additional conditions alongside the input noise vector. This allows for the generator and discriminator to be guided by the conditional input, so the data augmentation process is not completely stochastic. In our work, we seek to use the intact (non-masked) modality (i.e., posture or interaction data) to be the conditional input to the generator, so the generator imputes the missing modality based on the associated, non-missing data.

10.2 Conditional Variational Autoencoders

C-VAEs are similar to C-GANs with regard to the conditioning of a generative model (Sohn, Yan, & Lee, 2015). A variational autoencoder contains two neural network models, an encoder and a decoder, analogous to a standard autoencoder as described in Section 2.3.3. Likewise, the encoder learns to model latent variable representations of the input data, while the decoder reconstructs the original input based upon the generated latent representation. However, the VAE model constrains the latent representation to follow a specified probability distribution, typically a Gaussian distribution. Thus, the loss function of the VAE typically consists of two terms: one based on the reconstruction error and the other based on the Kullback-Leibler divergence between the two relevant distributions (i.e., the latent representation distribution and the Gaussian distribution). In the C-VAE implementation, the input of the encoder and the decoder are conditioned using the

same corresponding condition vector as that used with the C-GAN. In this work, we use the same intact modality as that which is the condition to the C-VAE model.

10.3 Model Evaluation (USMA)

We evaluate the multimodal data imputation process using the same dataset splits as those described in Chapter 9. To resolve class imbalances within each dataset, the SMOTE technique was used to upsample the minority class for each affective state. Automated feature selection was performed based on the training set by identifying the features with the highest chi-squared correlation with the binary emotion label. To combine the multimodal feature data, 15 features were selected from each modality, and concatenated at a feature-level to train each affect model. Following the feature selection process, five classification techniques were evaluated for each affective state: support vector machine, logistic regression, Gaussian Naïve Bayes, random forest, and feedforward neural network. The models were trained using 4-fold cross-validation while performing iterative grid search on the model hyperparameters to determine the optimal model, with the best model's performance reported using the held-out test set. During the cross-validation process, synthetic data resulting from upsampling was removed from each fold used as a validation set to avoid artificially inflated performance.

Each generative model was evaluated by masking either the interaction-based modality or the posture-based modality. To evaluate the performance of the generative models for varying levels of missing data, each modality was masked by selecting 25%, 50%, or 75% of the data points (BROMP observations) for each student. The posture data was masked intermittently throughout the student's session, with each data point having an equal probability of being masked. This was equivalent to masking posture data in 20-second intervals, as each BROMP observation coincided with a roll up of the prior 20 seconds of student behavior. By masking 20-second

intervals, we effectively simulated sporadic data loss, such as that caused by mistracking, a student exiting the sensor's field of view, or intermittent sensor error, while posture data was missing for consecutive readings. The interaction log data was masked by masking the last 25%, 50%, or 75% of the student's data. This resembled real-world situations in which an adaptive learning environment crashes or fails, resulting in data loss for the remainder of the student's session. The data was masked by setting all features in the masked modality to be missing for a selected data sample or sequence. The original values were stored as ground-truth data for evaluation of the data imputation methods. This masking process was performed on the training data described earlier for inducing affect detection models.

Following this phase, fully connected generative models were trained on the non-masked data. The C-VAE was trained using non-masked features from the masked modality as input, and the features from the intact modality are used as conditions for the model. The C-GAN also used features from the intact modality as the conditions, but the generator received a Gaussian noise vector of size 32 as input. The C-GAN's discriminator took an input of either "fake" data produced by the generator or "real" data consisting of non-missing data samples from the masked modality. The generative models are trained for 1,000 epochs each, with hyperparameter tuning performed on the number of layers in the generator and discriminator (C-GAN), the number of layers in the encoder and decoder (C-VAE), and the learning rate. Each generative model was optimized using the ADAM optimization algorithm (D. Kingma & Ba, 2017), and binary cross entropy as the loss function. Additionally, each model utilized a hyperbolic tangent activation function in the hidden layers, necessitating a normalization of the data to be within the range of -1.0 to 1.0.

The optimal model was determined by calculating the root mean squared error (RMSE) between the imputed values and the original, ground-truth values. The optimal model was then

used to impute the missing values in the training set, which was subsequently used to train the affect detection models. The changes in the affect models' predictive accuracy demonstrated how different data imputation methods approximate masked or removed predictive features or trends in the affect training data. In this way, the effectiveness of the generative data imputation methods was evaluated in two different thrusts. For comparison, we evaluated the generative models against two baseline methods: mean imputation and probabilistic matrix factorization (PMF) (Salakhutdinov & Mnih, 2008). Mean imputation was implemented by taking the mean value of each feature within a student's session data. PMF has become relatively common within recommender systems, in which data imputation is a frequently encountered task. A sparsely populated matrix is factored into two distinct lower-rank matrices, the multiplication of which approximates the original data. This process is repeated iteratively to minimize the reconstruction loss of the imputed data using expectation maximization.

10.4 Results (USMA)

To determine the impact that missing data and the resulting imputation have on affect detection, baseline models induced using multimodal data that was completely unmasked were evaluated. The ideal performance of the generative model would result in the imputed data matching the previously masked data samples perfectly, which would result in no deviation from the performance of the models trained on non-masked data. This allows the impact of the data imputation to be evaluated relative to the original, non-masked data, as well as to the performance of the affect models trained on masked and non-masked data.

We examined the predictive performance of the affect detection models trained with complete data using AUC, accuracy, precision, recall, and F1 score. The best performing classifier, hyperparameter configuration, and selected features based on non-masked data were then

preserved for the data imputation phase. This allowed for any deviation in the model's performance to be attributed to the data imputation process rather than other factors. For the purpose of this work, the affect detection models' predictive performance served as a "gold standard" for examining how different generative imputation methods impacted the performance of the affect models trained with missing data.

Table 10.1 shows that the feedforward neural network was the optimal classification model for each of the affective states. All of the affect models achieved an AUC greater than random chance (0.500) with the exception of *confused*. *Frustrated* and *surprised* had relatively low precision and recall values; this could be attributed to class imbalances, as positive instances of both classes individually comprised less than 5% of the total dataset.

Table 10.1. Performance of optimal affect models (USMA).

Emotion	Model	AUC	Accuracy	Precision	Recall	F1 Score
Bored	FFNN	0.837	0.840	0.395	0.833	0.536
Confused	FFNN	0.462	0.463	0.202	0.459	0.281
Engaged Concentration	FFNN	0.620	0.636	0.628	0.807	0.706
Frustrated	FFNN	0.638	0.759	0.128	0.500	0.204
Surprised	FFNN	0.594	0.877	0.118	0.286	0.167

The four imputation models were evaluated across the five affective states, each using three possible levels of missing data. The interaction-based modality was masked using the end of each sequence as the masking location to simulate the loss of all interaction log data following a software failure in the middle of a student's interaction with the game-based learning environment. The results of data imputation on the affect model training data are shown in terms of RMSE in Figure 10.1. The impact of data imputation on affect models using the optimal hyperparameter

configuration from each model in Table 10.1 is illustrated in Figure 10.2. The impact on the affect models' predictive performance is measured in terms of the absolute difference between the original affect models' AUC, and the AUC of the same model when trained on the imputed data. Both affect models are evaluated on the same held-out test set.

As shown in Figure 10.1, deep generative models yielded the best data imputation performance for four of the five affective states when 25% of the interaction log modality was masked, for two of the five when 50% was masked, and for three of the five when 75% was masked. Specifically, the C-VAE was the best performing model in two cases, while the C-GAN was the most effective imputation model in seven cases. However, C-GAN imputation appeared to have a less adverse impact on the affect detection models (Figure 10.2). From this perspective, the C-GAN was the best performing imputation approach for 12 of the 15 total evaluations, including all of the 25% missingness level. The C-VAE was the best model for only two of the 15 evaluations in terms of adverse impact on the affect models, both occurring with the *surprised* affective state. In total, data imputation with generative modeling had the least adverse impact on the affect detectors' performance in 13 of the 15 cases.

In a similar manner, the same four imputation models were evaluated using the posture data as the masked modality, while the corresponding interaction log data was used as the conditional modality for the generative models. The primary difference from the gameplay modality masking is that the posture data was masked using a uniform probability across the entirety of each student's sequence. For example, 25% of the posture data samples were selected to be masked, but that selection occurred with equal probability across all data samples. This was done to ensure realistic masking of the sensor-based modality by simulating intermittent issues that may occur throughout student interactions with an adaptive learning environment, such as

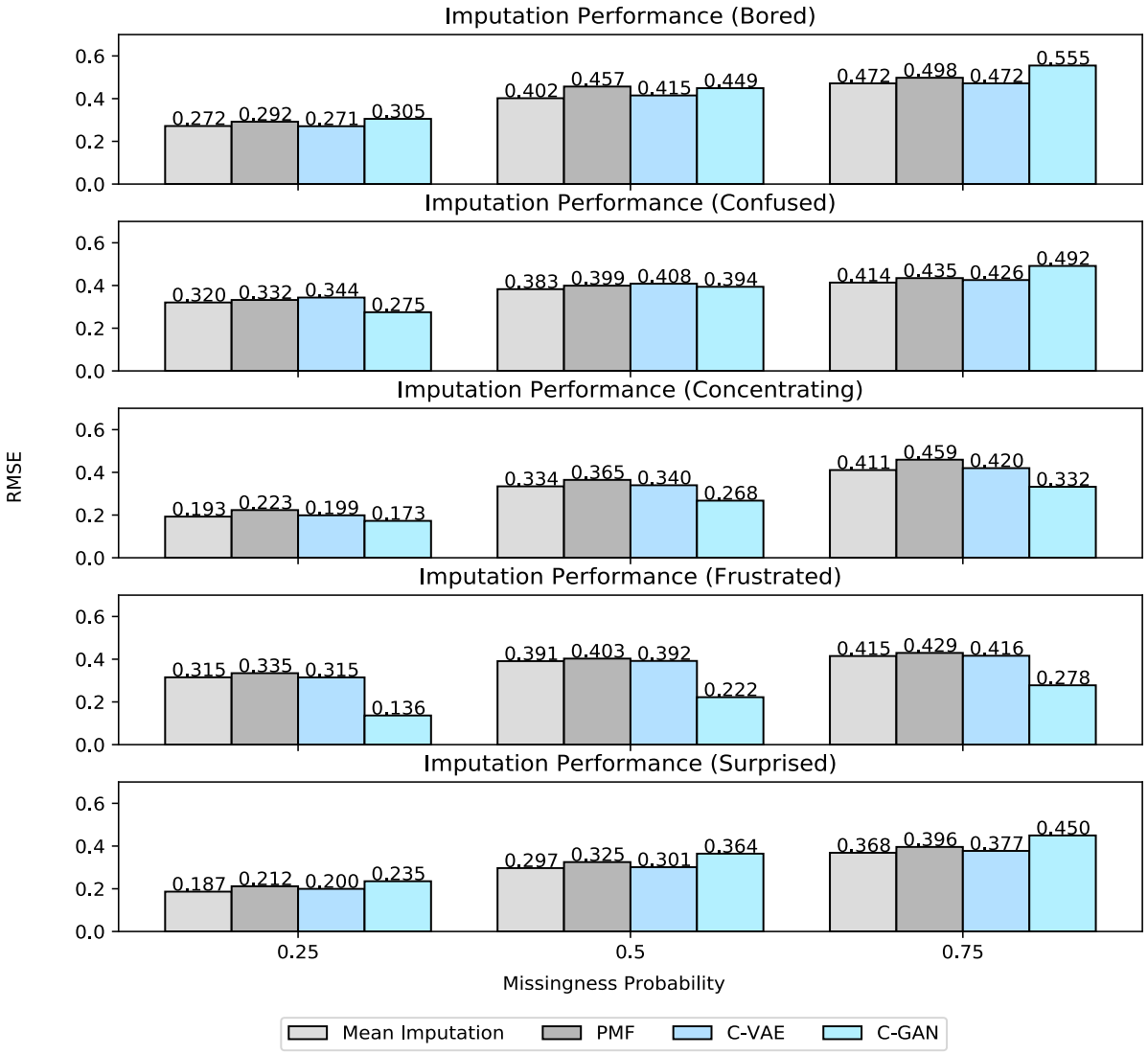

Figure 10.1. Imputation performance for the interaction log modality (lower is better).

sensor mistracking, noise, or reliability issues. The results of these evaluations are based on the missing posture data, evaluated for the same 25%, 50%, and 75% levels of missingness, and they are shown in terms of imputation performance (Figure 10.3) and impact on affect detector performance (Figure 10.4).

Generative models outperformed the two baselines in terms of imputation RMSE for four of five cases with 25% and 50% masking, and all five cases with 75% masking. The C-VAE was the optimal data imputation method in terms of RMSE for ten of the fifteen evaluations, compared

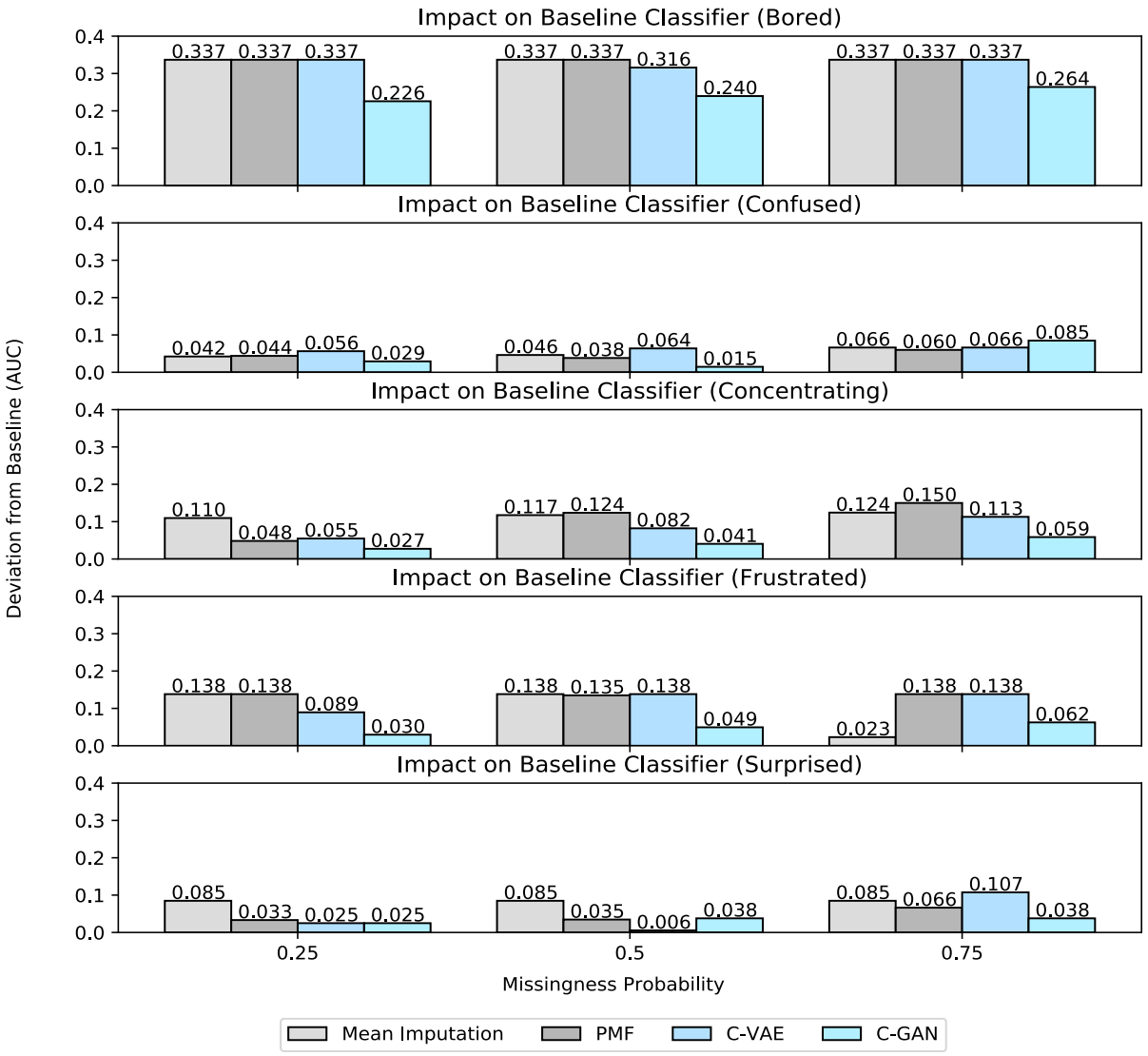

Figure 10.2. Impact of data imputation on affect models for interaction log modality (lower is better).

to only three for the C-GAN model. In terms of adverse impact on the affect models, the C-VAE was the optimal method in six cases, while the C-GAN was the optimal method for seven cases. However, in several cases the best-performing generative model was matched by one or both of the baseline models, such as when 50% of the *surprised* data was masked (Figure 10.4).

The findings suggest that multimodal conditional generative models outperform the two baseline data imputation methods in 60% (9 out of 15) of the interaction log masking evaluations across the three data missingness levels and five affective states with respect to data imputation

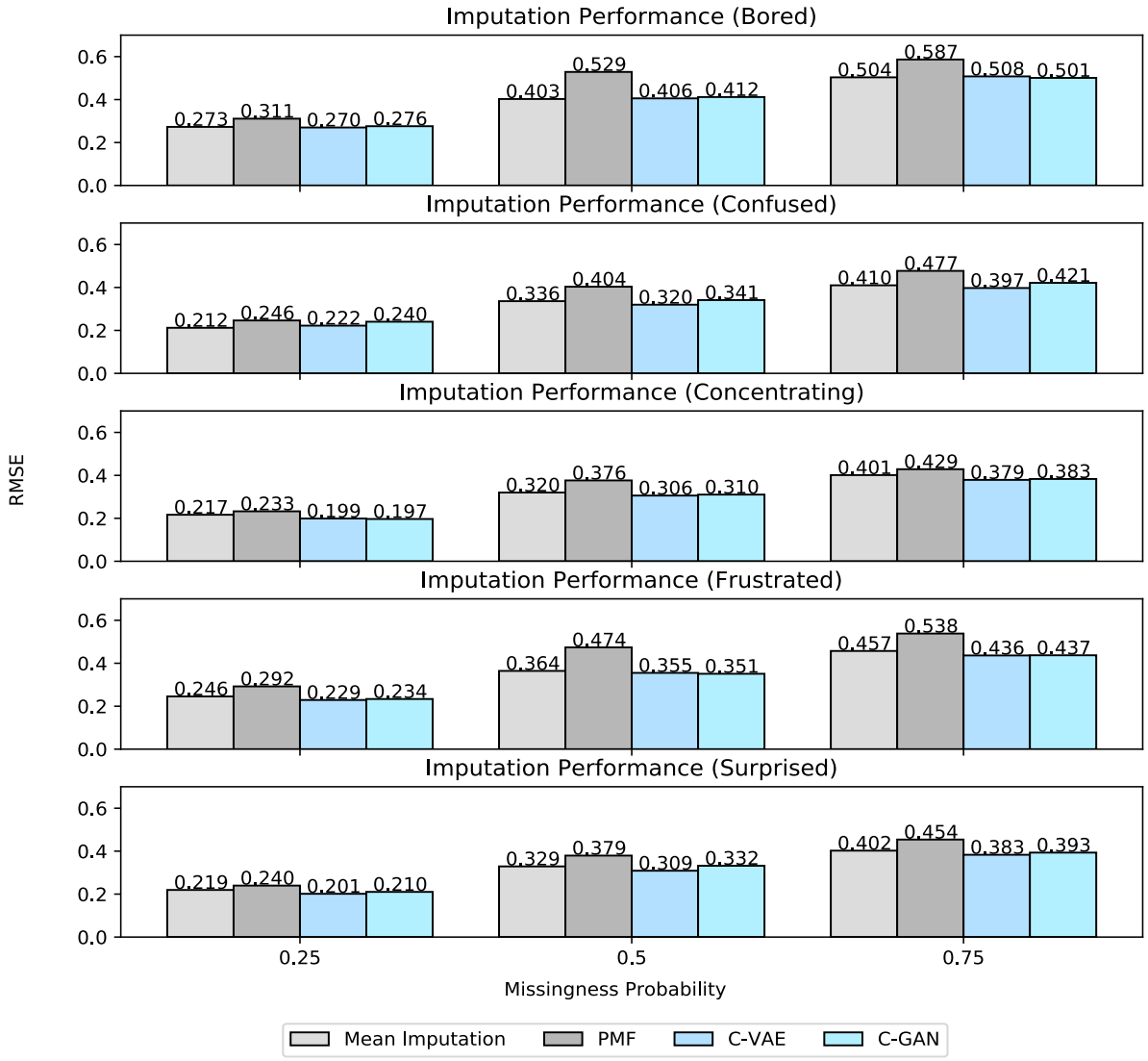

Figure 10.3. Imputation performance for the posture modality (lower is better).

RMSE. This is compared to generative modeling outperforming the baseline data imputation methods in 86.7% (13 out of 15) of the evaluations when using the posture data as the masked modality. However, in terms of mitigating the adverse impact of missing data on affect detection models' performance, generative models performed optimally for 86.7% (13 out of 15) of the total evaluations for interaction log masking, while posture masking resulted in generative models outperforming the baselines in 80% (12 out of 15) of the evaluations, indicating consistent performance across the two modalities. Although different imputation methods yielded the best

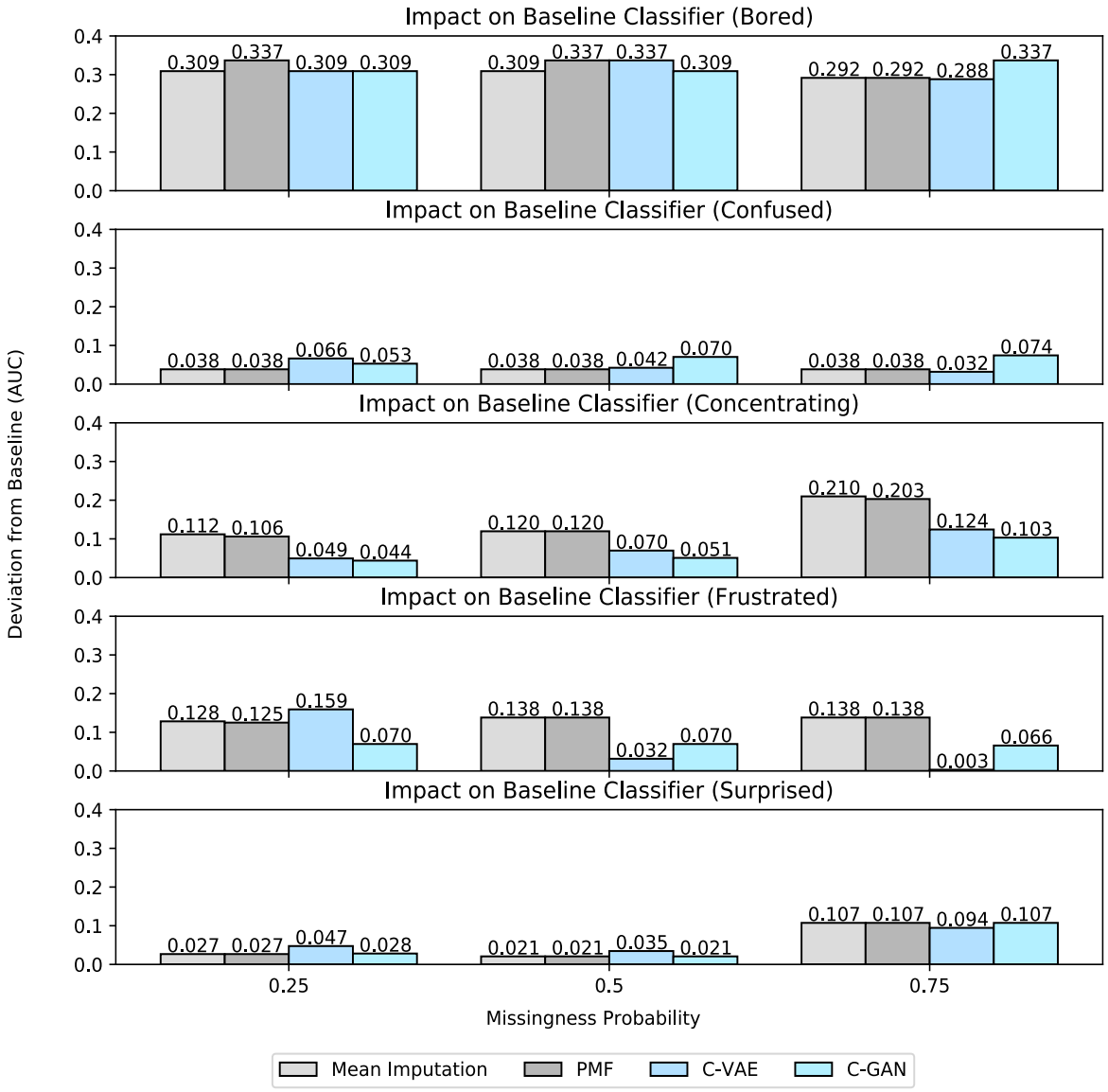

Figure 10.4. Impact on affect model performance for posture modality (lower is better).

performance for different affective states and modalities, it should be noted that different generative models could be utilized depending on the affective state and modality in a run-time setting. For this reason, we focus on the performance of deep conditional generative models as a family rather than individual models.

The generative models offer several benefits that contribute toward their higher performance than that of the baseline data imputation techniques. Because the generative models

are conditioned on separate, concurrent modalities, it is possible to maintain a multimodal perspective during the modeling process. This allows the imputation to be based on both non-missing data from the masked modality and on the other intact modality. Mean imputation takes into account only a single predictive feature, and the PMF model focuses only on a single modality during its imputation. Additionally, because the C-GAN and C-VAE are based on deep neural networks, they are well suited to extract and model complex patterns between the multimodal data that may otherwise be ignored or removed. By using a deep learning-based imputation approach, these underlying features are able to be partially or fully reproduced within the masked modality, which can prove beneficial to the predictive performance of affect detection models. This is a possible explanation as to why the conditional generative models are the optimal imputation method for at least 80% of the affect models examined.

It is notable that mean imputation appears to produce similar RMSE values compared to the C-VAE and C-GAN models for imputing missing data. Many of the features of the interaction log data were reported using either a standard deviation or an average number of certain gameplay actions. Because of the length of gameplay sessions (approximately one hour each), the variance of these attributes may be less than that for the posture-based data. Due to the averaging of these features within mean imputation, it is possible that trends in the interaction log data that may be of use to the affect models are smoothed during the imputation stage. This may explain why the mean imputation produced a RMSE similar to that produced by the generative models when the interaction log modality is masked.

It is notable that the C-GAN was the most frequent optimal imputation method in terms of RMSE for the interaction log data masking, while the C-VAE was the most frequent for the posture data. We observe that the average difference between the C-GAN and C-VAE's RMSE is 0.076

for masked interaction logs, and 0.010 for masked posture data, indicating that while the C-VAE may have outperformed the C-GAN more frequently for the posture-based evaluations, the margin between the two generative models was extremely slim. However, this does not appear to be the case with the interaction log masking, where the C-GAN outperformed the C-VAE (and baselines) by considerable margins.

While the two deep conditional generative methods showed improved data imputation performance in most cases, it should be noted that the two modeling techniques are inherently different: GANs are constructed for generative tasks through the adversarial setup of their architecture, whereas VAEs are primary intended for latent representation modeling by minimizing the loss defined with the Kullback–Leibler divergence and the reconstruction error. It is often a challenge to accurately contextualize or quantify GAN convergence or performance as a whole due to the competing situation between the generator and the discriminator. This motivates the need to extend the evaluation of data imputation techniques to consider adverse impacts on the affect models, which appears to provide additional support for the use of C-GANs as a multimodal data imputation method.

The performance of data imputation techniques appeared to be related to specific affective states. For example, during interaction log masking, the C-GAN produced the lowest RMSE for *frustrated* in each of the missingness levels but produced the highest RMSE for *surprised* in each of the missingness levels. This behavior was also observed when evaluating the variance of the affect models. This can be attributed to a number of factors, such as inherent data imbalances (particularly for *frustrated* and *surprised*), physical behavioral cues that are distinct for each affective state, and differently predictive features for each binary class label.

A decline in imputation performance is expected as the missingness level increases (Jaques et al., 2017; Yoon et al., 2018), particularly for deep learning models. While this behavior occurs for each of the affective states, the decline is not as drastic as might be expected given the size of the initial training data, step size of the masking (25% increments), and depth of the deep learning architectures. As the amount of intact data decreases for a certain modality, the generative models risk overfitting to the training data or generating random noise as output. However, this issue is partially mitigated through the conditional input to each of the generative models. The inclusion of the condition as input allows the discriminator to be provided with additional training data, consisting of 1) real, non-masked data with corresponding conditional data, and 2) fake, synthetic data generated with randomly selected conditional data. This allows for the generator to be further refined during the training phase and provides further support for the use of C-GANs within our multimodal data imputation framework.

10.5 Model Evaluation (*JavaTutor*)

The model evaluations using the *JavaTutor* corpus were similar to the experimental setup described in the previous sections. The masking intervals also remained the same, with posture-based feature vectors masked with an equal probability and with interaction-based and dialogue-based modalities being masked from a certain timestamp to the end. It should be noted that both the interaction-based and dialogue-based modalities were treated as a single modality regarding the artificial masking due to the fact that these two modalities originate from the same source, and it is assumed that in the case of software failure, both modalities would suffer from invalid or missing data.

The data processing (standardization, feature selection, model selection) pipeline remains the same as described in the *JavaTutor* evaluations from prior chapters, including the median split

that is used to convert the affect prediction task into binary classification tasks. The imputation and affect modeling components of the experiments are all conducted within each outer cross-validation fold within the 10-fold nested cross-validation process. The training of the C-VAE and C-GAN models remains the same as described in Section 10.3, with the removal of masked data occurring prior to training the generative models and the intact modality's features being used as conditioning features for each model. The hyperparameter tuning was performed across the number of nodes and number of hidden layers in the generator and discriminator components of the C-GAN, as well as the number of nodes and number of hidden layers in the encoder and decoder portions of the C-VAE. Additional hyperparameter tuning focused on the learning rate of the two generative models. The C-GAN was trained for a total of 100 epochs.

Probabilistic matrix factorization is retained as a baseline imputation method, but mean imputation is replaced with a more complex method, multiple imputation by chained equations (MICE) (Van Buuren & Groothuis-Oudshoorn, 2011). The MICE algorithm entails imputing the missing values of each feature using mean imputation, and then using the imputed values to perform a linear regression on each individual feature to predict the missing values for that particular feature. This process is performed across all features in a single iteration and continues iteratively until the imputation process converges.

10.6 Results (*JavaTutor*)

To provide a baseline of complete multimodal affect data without any artificial data masking, we train a series of affect models for each affect state, similarly to the work performed in Chapter 8, and in Section 10.4. The results from these evaluations (Table 10.2) serve as the "gold standard" that will be used to determine the deviation in predictive performance as a measure of the imputation's impact on the models' affect detection. Because we use a median split on the target

variables, we use F1 Score instead of AUC as the primary evaluation metric. Following this process, the optimal model's hyperparameter values for each affective state are retained and used to predict affect using the artificially masked and subsequently imputed multimodal dataset.

Table 10.2. Performance of optimal affect models (*JavaTutor*).

Emotion	Model	F1 Score	Accuracy	AUC
Engagement	FFNN	0.652	0.563	0.554
Frustration	FFNN	0.678	0.532	0.548

Similar to the work described in Section 10.4, the four imputation models were evaluated across the two affective states, each using the same three possible levels of missing data. The interaction-based and dialogue-based modalities were masked using the end of each sequence as the masking location to simulate the loss of all interaction log data following a software failure in the middle of a student's interaction with the game-based learning environment. The results of data imputation on the affect model training data are shown in terms of RMSE in Figure 10.5. The impact of data imputation on affect models using the optimal hyperparameter configuration from each model in Table 10.2 is illustrated in Figure 10.6. The impact on the affect models' predictive performance is measured in terms of the absolute difference between the original affect models' F1 Score, and the F1 Score of the same model when trained on the imputed data. All affect models are evaluated using the same student-level cross-validation folds to ensure consistency across experiments.

Deep generative models yielded the best data imputation performance for each of the six evaluations across the two affective states and three masking levels. It is noted that the C-VAE was the lowest-performing model in terms of imputation accuracy, with the lone exception of the

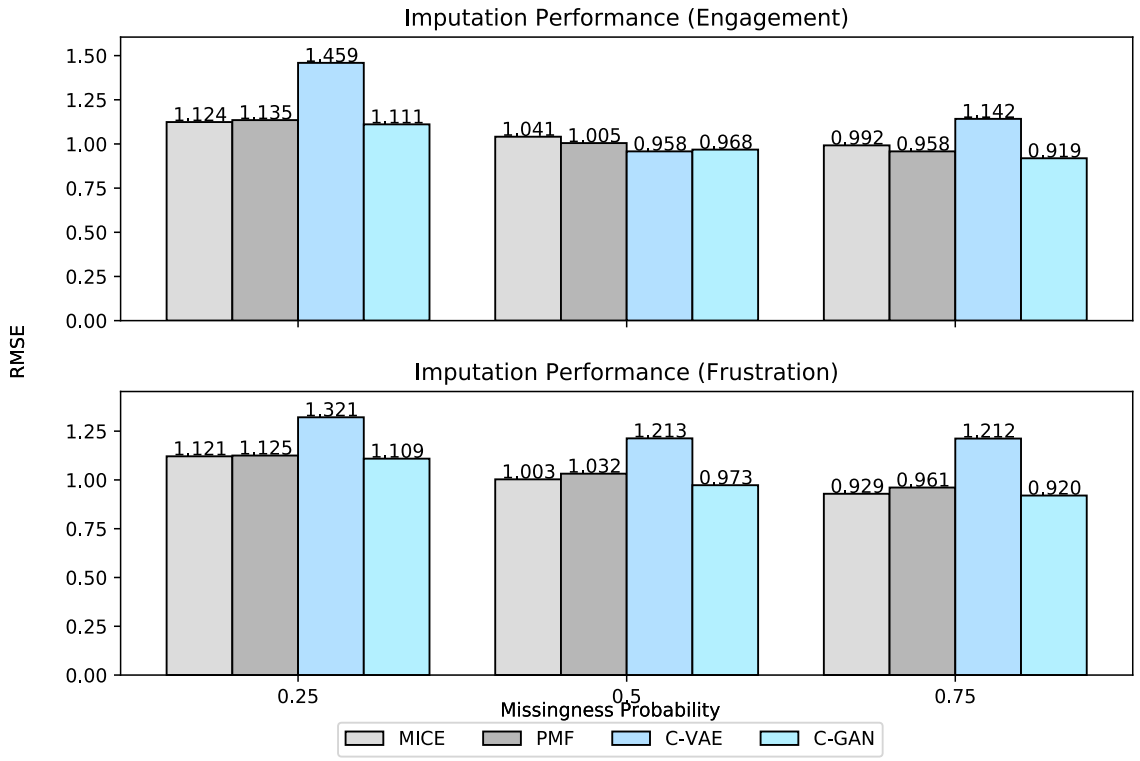

Figure 10.5. Imputation performance for interaction and dialogue modalities (lower is better).

50% masking for *engagement*. For the other evaluations, the C-GAN outperformed all of the other imputation models, although in most cases, the improvements were marginal. In two cases, the MICE imputation and the C-GAN imputation were relatively similar, and the only imputation model that showed distinct performance was the C-VAE for five of the six evaluations, where the imputation performance was significant worse than the C-GAN and baseline models.

In terms of the impact on the affect detection models for the interaction and dialogue masking, the generative imputation models achieved the highest performance for all six experiments, with the C-VAE being the optimal model for five of the evaluations, and the C-GAN showing optimal performance for the remaining evaluation. However, it is noteworthy that the C-VAE was the highest-performing imputation model in terms of the F1 Score deviation but was frequently the lowest-performing imputation model in terms of RMSE. This points to a potentially

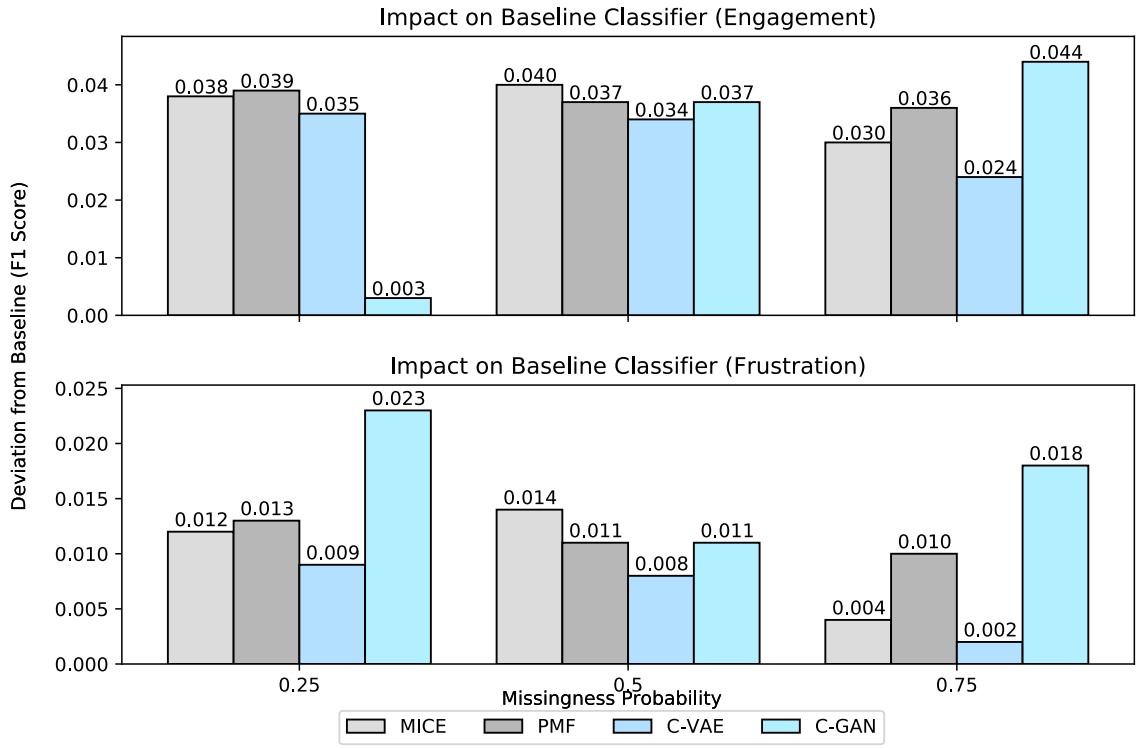

Figure 10.6. Impact on affect model performance for interaction and dialogue modalities (lower is better).

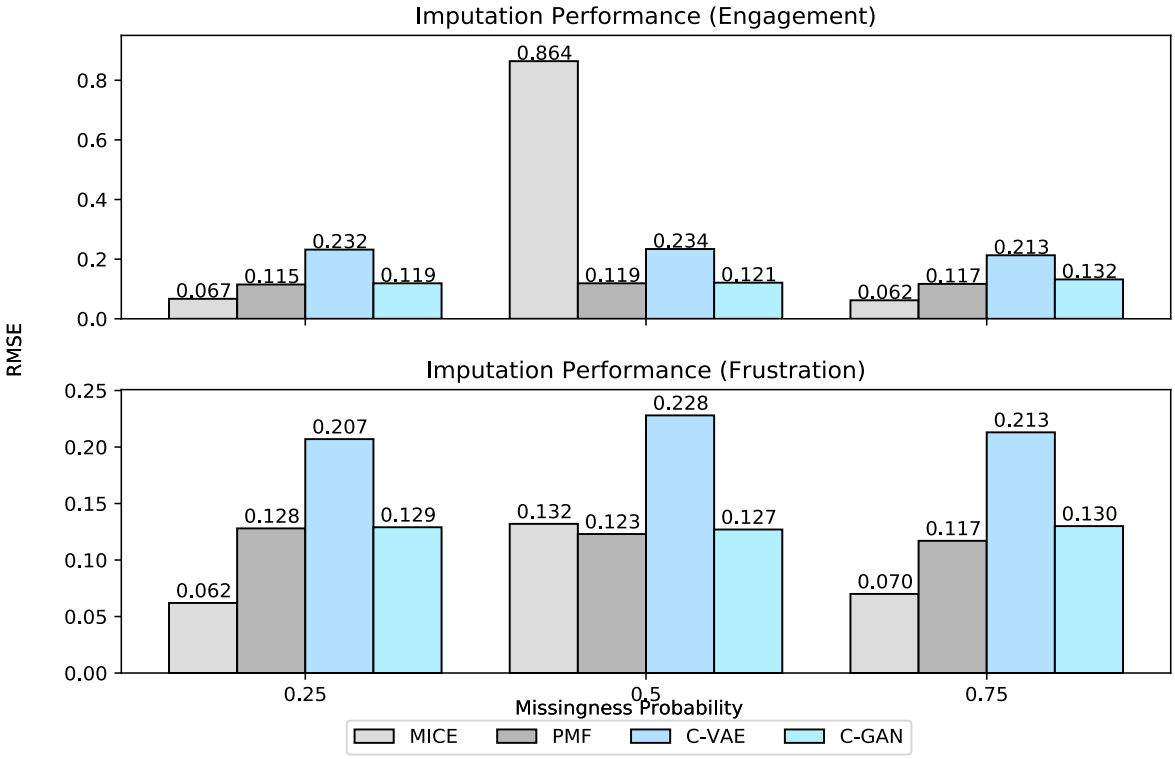

Figure 10.7. Imputation performance for the facial expression modality (lower is better).

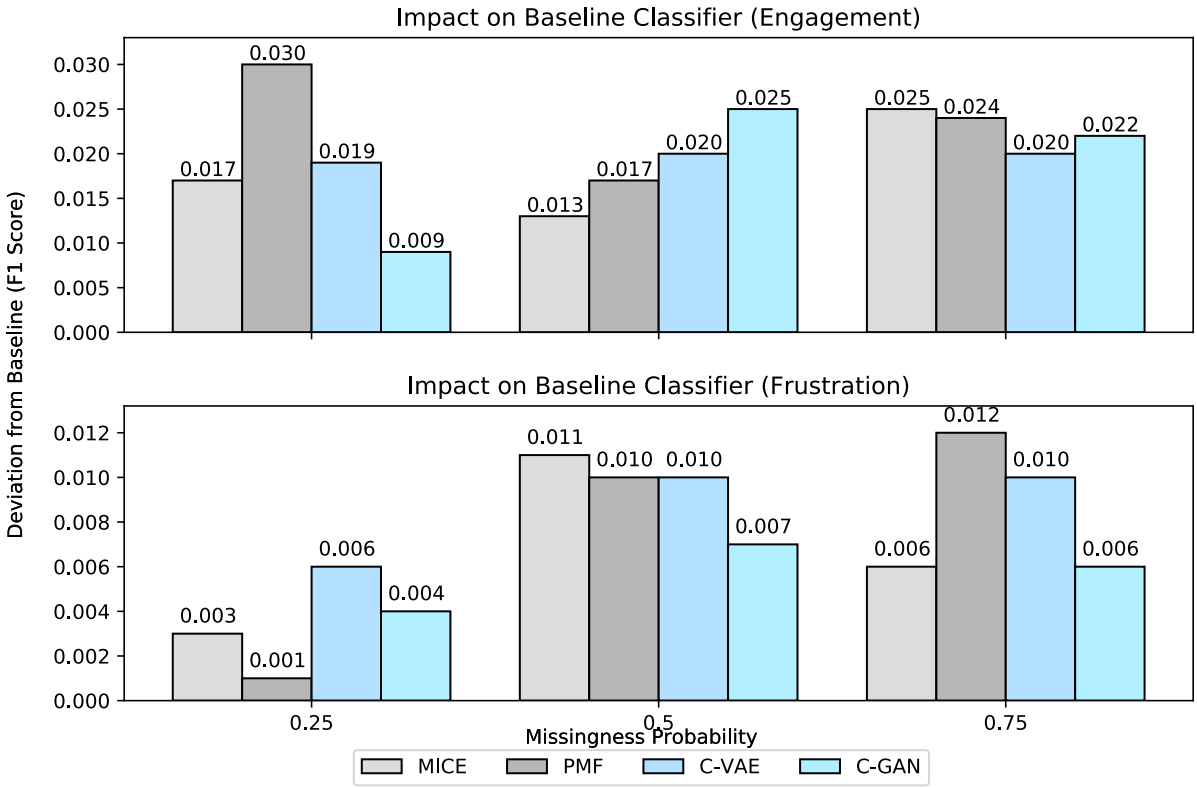

Figure 10.8. Impact on affect model performance for the facial expression modality (lower is better).

non-linear relationship between raw imputation performance and the predictive performance of the affect models on the imputed multimodal data.

Following these evaluations, we apply artificial masking to the facial expression data and retain all values in the interaction and dialogue modalities with the same experimental setup mentioned previously. The imputation performance is shown in Figure 10.7 and the impact on the affect models' classification performance in terms of F1 Score deviation is shown in Figure 10.8.

For the facial expression masking, the deep generative models did not reach optimal performance for any of the six evaluations, with MICE being the optimal imputation method for 2/3 of the evaluations for *engagement* and 2/3 of the evaluations for *frustration*. The exceptions were the PMF-based imputation for the 50% masking for both affective states. Similar to the previous experiments, the C-VAE frequently returned the highest imputation error in terms of

RMSE, while the C-GAN performed relatively well, but was never the optimal imputation model. This may point to the fact that the C-VAE may not serve as a generalizable method for data imputation in terms of raw imputation accuracy, and the C-GAN achieves more accurate performance in that regard. In terms of the F1 Score deviation, the generative models reached the highest performance for four of the six evaluations, the C-GAN for 3 (tie with MICE for one experiment) and the C-VAE for 1. The generative models appeared to perform stronger on the *frustration* classification tasks compared to the *engagement* classification tasks. In total, the generative models were the optimal imputation model for 16 of the 24 experiments, although in some cases, the improvement was marginal. Additionally, the generative imputation method did not achieve optimal performance for any of the facial expression masking experiments for *engagement*. As a result, we conclude that the deep generative modeling approaches show promise within different contexts, but additional evaluations are needed to confidently conclude that these models' imputation capabilities flexibly generalize to different modalities and datasets.

CHAPTER 11

MULTIMODAL AFFECT MODELING WITH MULTI-TASK LEARNING

In this chapter, we investigate the integration of temporal contextual features from students' affective sequences in the USMA dataset as a means to improve models of student affect through multi-task learning. While a significant body of prior work focuses on predicting individual occurrences of various affective states (Henderson et al., 2020b; Jiang et al., 2018), these approaches often ignore the predictive information offered by a student's overall affective trajectory, such as how a student transitions from one affective state to another throughout a single learning session. The shifts in a student's affective states have been shown to reveal particular recurring patterns, and thus can provide predictive value in affect modeling. We hypothesize that using students' future affective states as they engage with a game-based learning environment can be utilized as an auxiliary multi-task function to improve the predictive performance of the affect detection models. Additionally, we explore how to optimally combine interaction-based and posture-based modalities by exploring potential deep learning architectures: multi-task fully connected feedforward neural networks and cross-stitch neural networks. Finally, we examine the benefit of including a student's prior affective states as a means of providing additional affect sequence information to improve the performance of affect detection models. Our results indicate that the use of MTL to model affective sequence patterns from each student leads to improved prediction of multiple affective states, and the use of cross-stitch neural networks further strengthens predictive accuracy. Because there are not multiple affect labels for a single trajectory in the *JavaTutor* corpus, we evaluate the same neural architectures and MTL approaches by predicting multiple affect states simultaneously, another potential application of MTL within deep learning for affect detection.

11.1 MTL with Affect Sequences

To adapt the single-task affect detection approach to an MTL formulation, the target variables were expanded to include a one-hot representation of each possible affective state. The one-hot vector was indicative of the affective state B_{i+1} that followed the current BROMP observation B_i (Figure 11.1). B_i was a binary indicator of the presence of one of the five possible affective states. Therefore, the multi-task models were modeled using a label vector of size 6 (binary indicator of a single affective state + one-hot vector of size 5). Using the affect model for bored as an example, the multi-task output vector for a positive annotated occurrence of bored followed by an annotation of confused would be [1, 0, 1, 0, 0, 0], while a negative annotated occurrence of bored followed by a subsequent annotation of frustrated would be [0, 0, 0, 0, 1, 0] (Henderson et al., 2021b).

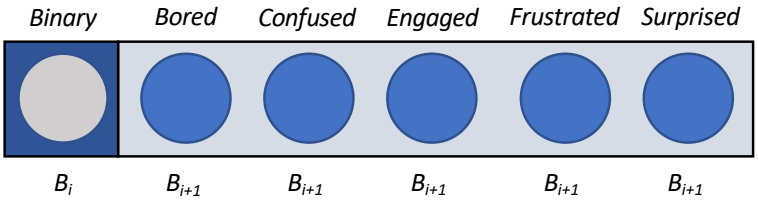

Figure 11.1. Multi-task feature vector representation.

Because the multi-task models are predicting future occurrences of each affective state, it is impractical to utilize these labels as input features as this information would not be available in a run-time environment. As a result, we use these labels as auxiliary output variables for the purpose of boosting the predictive performance of the multi-task models relative to the current affective state, an approach that has been previously demonstrated to improve predictive performance (Caruana & De Sa, 1996; Liu et al., 2015). This process can be employed when certain features are unhelpful for predicting other output variables or are not available until after the predictions are made, allowing the features to be used to present additional information to the

model during the training process only (Caruana & De Sa, 1996). In this case, presenting the subsequent affective state to the multi-task model allows the model to potentially observe differential patterns in student behavior prior to transitioning to another affective state. For example, a student's postural behavior while currently in a state of engaged concentration may fluctuate depending on if the subsequent affective state is also engaged concentration or a different state such as confusion. By introducing additional predictive tasks, the model is trained to extract temporal features and patterns from affective sequences that can improve the model's prediction of the current affective state. The occurrences of each two-step affective sequence are shown in Figure 11.2.

The most common affect sequences are persisting states of engaged concentration (denoted as "Concentrating" in Figure 11.2), consecutive states of confusion, and alternating between these two states. This result aligns with the proposed model by D'Mello and Graesser (2012). Other notable sequences are students' transitions between states of bored and engaged concentration, particularly as this indicates that students are often capable from returning to an engaged state while previously being in a state of relative disengagement, a behavior previously observed by Andres et al. (2019).

11.2 Cross-Stitch Networks

An active area of investigation in multi-task deep learning is determining the appropriate level of layer connectivity across each task. A multi-task model that consists of only fully connected layers contains the highest level of connectivity, as each layer propagates the same fully shared data representation across all tasks with the exception of task-specific output layers. Alternatively, to avoid any inter-task communication within the multi-task framework, separate models can be trained for each task, so that the trained weights are unique for each output. While prior work

applying multi-task learning for student modeling utilizes full connectivity across tasks within the model's hidden layers (Geden et al., 2020; Henderson et al., 2020c), other work within computer vision has explored the benefits of "split" neural architectures, or architectures that maintain a degree of separation between tasks within a pre-determined subset of the model's hidden layers.

Cross-stitch networks were proposed by Misra et al. (2016) as a generalizable approach to implementing "split" architectures by implementing parameterized linear combinations between a network's hidden layers that can learn optimal weightings between shared and task-specific latent representations. This approach allows feature representations to be combined within certain hidden layers and shared across tasks while also maintaining separation between task-specific

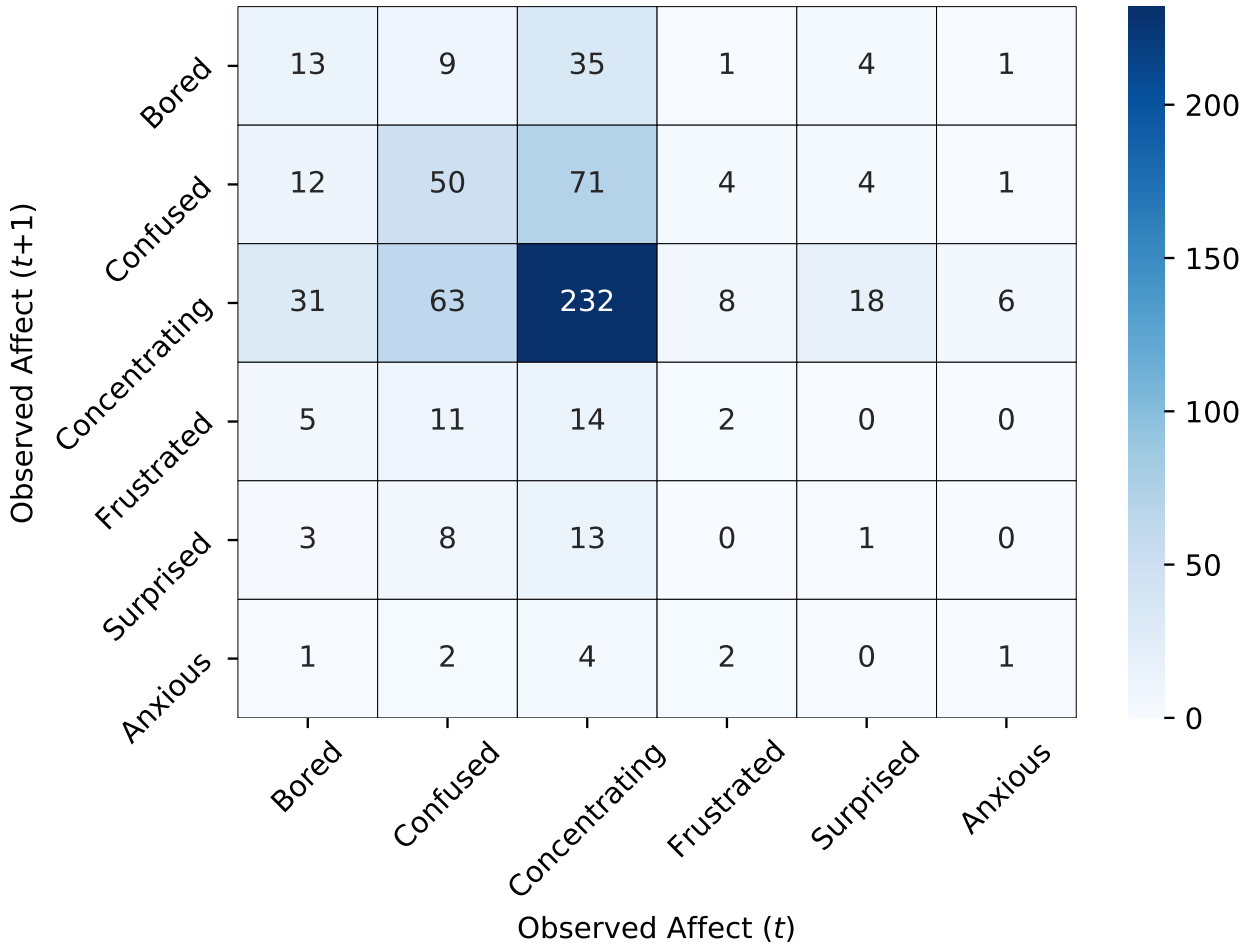

Figure 11.2. Frequency of each affective state and corresponding subsequent state.

representations. For example, in the case of modeling two tasks (A and B), a learned weight matrix α is used to parameterize the linear combinations of multiple tasks (α_{AB}, α_{BA}) as well as activations from a single task (α_{AA}, α_{BB}) (Figure 11.3). A value of 0.5 for α indicates that the representations are equally shared, with a value of 0 or 1 indicating that the representations are completely separate. Specifically, Equation 1 shows how the shared representation $\hat{x}$ is calculated at row i and column j by a cross-stitch unit that takes an input activation map, x:

$$\begin{bmatrix} \hat{x}_A^{i,j} \\ \hat{x}_B^{i,j} \end{bmatrix} = \begin{bmatrix} \alpha_{AA} & \alpha_{AB} \\ \alpha_{BA} & \alpha_{BB} \end{bmatrix} \begin{bmatrix} x_A^{i,j} \\ x_B^{i,j} \end{bmatrix} \quad (1)$$

The values of the weight matrix α are adjusted during backpropagation, with the partial derivatives easily calculable as the cross-stitch units are modeled with linear combinations. We evaluate cross-stitch networks alongside a multi-task variant of fully connected feedforward neural networks in our modeling of student affect to investigate whether the level of connectivity within each architecture has an observable impact on the predictive performance of the affect models.

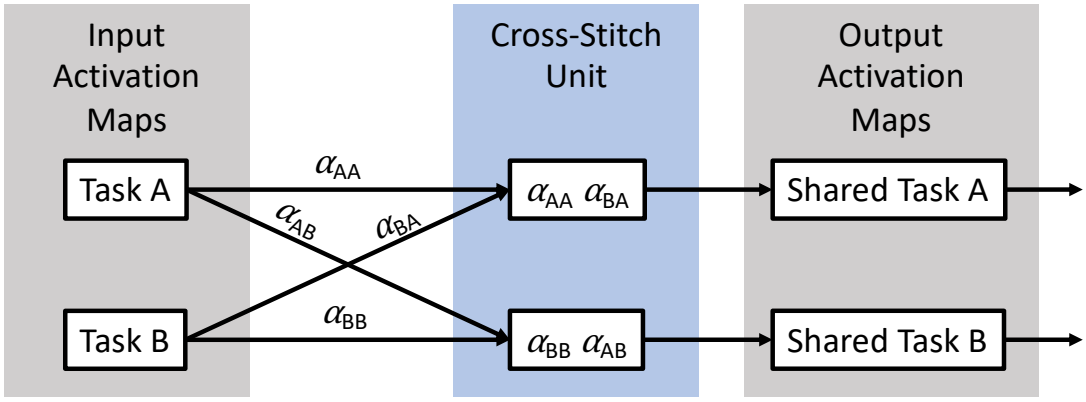

Figure 11.3. Visualization of a cross-stitch network for weighting shared representations between task *A* and task *B*.

11.3 Affect Model Evaluation (USMA)

To evaluate the performance of the multi-task models (fully connected and cross-stitch networks), we train a series of single-task neural and non-neural baseline models in addition to several non-neural multi-task baseline models. The baseline models were *k*-nearest neighbor, elastic net, random forest, and feed-forward neural network. These were selected as baselines due to their capabilities of both single-task and multi-task learning. The single-task baseline models demonstrate the performance of models without any affective dynamics context, while the multi-task non-neural baseline models verify that the deep learning-based approaches (fully connected and cross-stitch networks) achieve higher performance with the affective dynamics context than non-neural multi-task models.

Each model was evaluated with nested ten-fold cross-validation, with each fold split at a student-level to prevent data leakage across the training, validation, and test sets. Within each outer cross-validation fold, the data were standardized to ensure a mean of zero and standard deviation of one prior to performing feature selection. Hyperparameter tuning was performed using three-fold cross-validation within the training data of the nested outer cross-validation. The hyperparameters evaluated were the number of nearest neighbors (*k*-nearest neighbors), ratio of *L1* and *L2* regularization (elastic net), number of estimators (random forest), and the number and size of the hidden layers (neural network). Each deep learning model's hidden layer used a hyperbolic tangent activation function due to the standardization of the data, as well as a dropout probability of 0.5 in the last hidden layer to mitigate potential overfitting. The loss function for the feedforward single-task network was binary cross entropy. Additionally, minority cloning was employed as an oversampling technique to resolve the class imbalance present within each

affective state's dataset. This process clones each instance of the minority class until the class distribution is brought to a more uniform level.

An active line of investigation in MTL is determining the optimal distribution of the loss term across the different tasks. A common naïve approach to MTL loss is to assign a uniform weight to the loss term for each individual task *t* when calculating the summative loss term:

$$L_{total} = \sum_t W_t L_t \qquad (2)$$

However, as the auxiliary tasks of predicting future affective states is distinguishable from the task of predicting the current affective state, we explore the use of a loss function that uses uncertainty weighting for each individual task (Kendall, Gal, & Cipolla, 2018). The weight W_t for each task is determined by maximizing the log likelihood of an assumed multivariate Gaussian distribution. By optimizing for the model parameters θ and observation noise σ, the following loss function is derived:

$$L_{total} = \sum_t \frac{1}{2\sigma_t^2} L_t(\theta) + \log(\sigma_t) \qquad (3)$$

In this way, optimizing for σ_t for each task *t* allows the relative weight of each task-specific loss function (i.e., the first term in Equation 3) to be learned from the data during the training process, while the second term in Equation 3 acts as a regularization term to prevent σ from increasing exponentially, which prohibits the model from learning. This allows the model to assign different weighted losses between the primary task (predicting the current affective state) and the secondary auxiliary tasks (predicting the subsequent affective state).

In addition to the single-task baseline model, four multi-task deep learning models were evaluated, uniformly weighted and uncertainty-weighted fully connected networks and cross-stitch

networks. Each deep learning model was trained for 100 epochs, with early stopping implemented using the validation set and a patience of 10 epochs. Each network contained either two or three hidden layers with each layer containing either 8, 16, 32, or 64 nodes. For each cross-stitch model, each pair of hidden layers contained a cross-stitch unit. Data standardization, feature selection, and minority cloning occurred within each outer cross-validation iteration using the training folds to protect against data leakage across the validation and test folds. The feature selection process followed a similar randomized, iterative forward feature selection process that is described in Section 7.4 for the *JavaTutor* corpus. Each nested cross-validation fold was kept consistent across all evaluations to ensure fair comparisons between models.

11.4 Results (USMA)

While the five auxiliary predictive tasks (i.e., predicting the subsequent affective state) were utilized within the training process, the model evaluation focused exclusively on the predictive performance of the current affective state only. As a result, the predicted values of the auxiliary affective states are not considered in the results presented in this section. The primary evaluation metric is Area Under Curve (AUC), due to its ability to account for data imbalances. Additional evaluation metrics are the raw accuracy and the F1 score for each model. For each affective state (*bored, confused, engaged concentration, frustrated, surprised*), the highest performing single-task baseline models in addition to all multi-task model variants are shown in Table 11.1, with the best performing models in terms of AUC highlighted in bold.

The feedforward neural networks (FFNNs) outperformed all other non-neural baselines (single-task and multi-task) across all affective states. Of note is the fact that the uncertainty-weighted models (annotated with "(W)" in Table 11.1) and the cross-stitch models are all variations of FFNNs, therefore FFNNs will always be the "optimal" model. The inclusion of subsequent

affective states as auxiliary outputs for multi-task modeling appeared to offer improved performance over the single-task baselines for all affective states. The uniformly weighted fully connected model was the highest performing affect model for *bored*, while the uniformly weighted cross-stitch model was the highest performing model for the remaining four affective states. Of note is the fact that, with the exception of *bored*, both variations of the fully connected model failed to outperform the single-task baselines, while the uniformly weighted cross-stitch network outperformed each baseline, and the uncertainty-weighted cross-stitch network outperformed the baseline in two affective states (*confused* and *surprised*). It did not appear that the uncertainty-weighted loss function improved performance significantly for either neural network model, achieving lower performance than almost every uniformly weighted counterpart. Although a multi-task model outperformed the single-task baseline for each affective state, the improvement was marginal at best for three affective states (*confused*, *engaged concentration*, and *frustrated*), with an incremental improvement of less than 0.02 in terms of AUC. The inclusion of the auxiliary affect sequence data showed improved performance for *surprised* and *bored*, which is surprising as the affect sequences for these two affective states typically transitioned to a state of *engaged concentration* (Figure 22). This indicates that a student's behavior while *bored* or *surprised* may differ depending on if the student is about to transition to a state of *engaged concentration* vs. a state of *bored*, *confused*, or another uncommon affective transition.

To further investigate the impact of integrating temporal information into the multi-task affect detection models, we include additional input features that represent the prior affective states exhibited by each student. This information is incorporated through five summative features representing the total number of observations of each of the five affective states prior to the current BROMP observation. The current BROMP observation is not included in these features as this

Table 11.1. Model evaluation results for single-task and multi-task models (USMA).

Bored			
Model Type	**AUC**	**Accuracy**	**F1 Score**
Single-Task	0.779	0.833	0.414
Fully Connected	**0.844**	**0.798**	**0.429**
Fully Connected (W)	0.839	0.848	0.414
Cross-Stitch	0.838	0.832	0.465
Cross-Stitch (W)	0.818	0.844	0.452
Confused			
Model Type	**AUC**	**Accuracy**	**F1 Score**
Single-Task	0.546	0.570	0.301
Fully Connected	0.528	0.752	0.009
Fully Connected (W)	0.508	0.722	0.110
Cross-Stitch	**0.563**	**0.620**	**0.313**
Cross-Stitch (W)	0.561	0.614	0.314
Engaged Concentration			
Model Type	**AUC**	**Accuracy**	**F1 Score**
Single-Task	0.586	0.554	0.589
Fully Connected	0.584	0.584	0.680
Fully Connected (W)	0.570	0.580	0.692
Cross-Stitch	**0.592**	**0.576**	**0.652**
Cross-Stitch (W)	0.561	0.586	0.666
Frustrated			
Model Type	**AUC**	**Accuracy**	**F1 Score**
Single-Task	0.594	0.742	0.115
Fully Connected	0.537	0.478	0.046
Fully Connected (W)	0.582	0.360	0.083
Cross-Stitch	**0.602**	**0.800**	**0.095**
Cross-Stitch (W)	0.592	0.789	0.117
Surprised			
Model Type	**AUC**	**Accuracy**	**F1 Score**
Single-Task	0.576	0.846	0.062
Fully Connected	0.523	0.702	0.055
Fully Connected (W)	0.507	0.624	0.078
Cross-Stitch	**0.646**	**0.622**	**0.099**
Cross-Stitch (W)	0.578	0.874	0.051

would be a form of data leakage. This process is the natural next step in incorporating affective dynamics within each student model, so that each model can be induced using both the antecedent and subsequent affective states. This allows the model to be trained using bidirectional affective

Table 11.2. Model evaluation results for single-task and multi-task models (with prior affective state features) (USMA).

Bored			
Model Type	**AUC**	**Accuracy**	**F1 Score**
Single-Task	0.779	0.833	0.414
Fully Connected	0.843	0.812	0.430
Fully Connected (W)	0.841	0.820	0.434
Cross-Stitch	0.854	0.811	0.435
Cross-Stitch (W)	**0.861**	**0.808**	**0.427**
Confused			
Model Type	**AUC**	**Accuracy**	**F1 Score**
Single-Task	0.546	0.570	0.301
Fully Connected	0.559	0.618	0.323
Fully Connected (W)	0.551	0.613	0.304
Cross-Stitch	0.560	0.622	0.327
Cross-Stitch (W)	**0.614**	**0.635**	**0.368**
Engaged Concentration			
Model Type	**AUC**	**Accuracy**	**F1 Score**
Single-Task	0.586	0.554	0.589
Fully Connected	0.597	0.619	0.685
Fully Connected (W)	0.606	0.601	0.658
Cross-Stitch	**0.627**	**0.604**	**0.676**
Cross-Stitch (W)	0.614	0.608	0.679
Frustrated			
Model Type	**AUC**	**Accuracy**	**F1 Score**
Single-Task	0.594	0.742	0.115
Fully Connected	**0.665**	**0.600**	**0.073**
Fully Connected (W)	0.633	0.360	0.064
Cross-Stitch	0.572	0.784	0.057
Cross-Stitch (W)	0.648	0.780	0.123
Surprised			
Model Type	**AUC**	**Accuracy**	**F1 Score**
Single-Task	**0.576**	**0.846**	**0.062**
Fully Connected	0.537	0.823	0.037
Fully Connected (W)	0.507	0.762	0.031
Cross-Stitch	0.548	0.685	0.070
Cross-Stitch (W)	0.526	0.831	0.047

sequences. The same model architectures using the future affective states as auxiliary tasks for the multi-task models shown in Table 11.1 are re-evaluated while also incorporating the prior affective

states as input features (Table 11.2). The same single-task baseline results are included in Tables 11.1 and 11.2 because the purpose of these models is to demonstrate the affect models' predictive performance without providing *any* affective dynamics context.

The addition of the preceding affect information into the input features increased the performance of the affect models for four affective states, with the exception of *surprised*. Additionally, the uncertainty-weighted loss function induced the highest performance for two affective states, *bored* and *confused*. Among the four affective states, the fully connected multi-task model was the highest performing model for one affective state, with the rest being modeled most effectively by cross-stitch networks. This provides further evidence for the enhancements offered by dynamically weighting the balance between the shared and task-specific representations within the multi-task models.

To investigate whether the improvements of the multi-task affective dynamic-based models are attributed to random chance, the results of the single-task baseline models are compared with the results of the optimal multi-task model for each affective state. The cross-validation results of the models were compared using a Wilcoxon signed-rank test, which is a non-parametric statistical test. This measure is used as the cross-validation results cannot be assumed to be normally distributed. Using a significance level of 0.05, the affective dynamics-based models were shown to demonstrate significant increases in performance for four affective states: *bored* (p=0.023), *confused* (p=0.018), *engaged concentration* (p=0.038), and *surprised* (p=0.038). Although the improvement of the affective models for *frustrated* was noticeable (0.071 AUC), it was not observed to be statistically significant (p=0.121).

There are limitations of this work that should be noted. The modeling of affective dynamics is reliant on having multiple labeled observations of affect for each student and is therefore

incompatible with affective training sets that contain only a single label per student. Additionally, annotated occurrences of affect are sometimes removed in cases of inter-rater disagreement, which results in gaps in students' affective sequences throughout their learning sessions. While using summative or averaged feature representations can help mitigate this issue, this approach removes any semblance of temporal order between multiple affect labels. Therefore, an area of future research is to investigate the tradeoff between temporal context and model performance. For the purposes of this study, we utilize the annotated observations of affect to represent the prior affective states for the affective dynamics-based models. However, the models may not have access to annotated observations of a student's prior affective states in a run-time setting. In these instances, the models would require an alternative representation of prior affect, such as the model's prior predictions or confidence intervals for each affective state. Finally, the predictive performance of this work is likely dependent on the number of observations of each affective transition present in the dataset. For example, roughly 30% of the possible affective transitions in our dataset contained less than two observations. While these transitions are less likely to occur in a real-world setting, this likely impacts the overall generalizability of the models.

11.5 Affect Model Evaluation (*JavaTutor*)

The experiments involving the *JavaTutor* corpus deviate from the affect sequence modeling in the prior sections due to the fact that a student only took a single post-test survey at the end of a learning session, which resulted in only one affect label per session. As a result, there are no affect sequences of multiple affective states across a single learning trajectory, which means we cannot evaluate the affect sequence modeling approach on this data corpus. However, to evaluate the generalizability of the MTL approach with the cross-stitch network architectures, we train the affect models to predict the affective states of *engagement* and *frustration* simultaneously; this

approach allows us to also evaluate the uncertainty-weighted loss function and the cross-stitch network approach.

The experiments described in this section use the same 10-fold nested cross-validation pipeline from the prior *JavaTutor* experiments in previous chapters, including the data standardization, feature selection, and hyperparameter tuning process. To evaluate the single- and multi-task capabilities of different models, we evaluate the same single-task baseline models from Section 11.3, including the *k*-nearest neighbor, elastic net, random forest, and feed-forward neural network models. In addition, the feed-forward neural network model is used as a multi-task baseline approach, in addition to the cross-stitch networks described in Section 11.2. All multi-task models are evaluated with uniformly weighted loss in addition to the uncertainty-weighted loss described in Section 11.4. The target variables were split using the threshold values from Section 9.6. To resolve the class imbalances, minority cloning was used for data upsampling.

11.6 Results (*JavaTutor*)

The results from the MTL evaluations are shown in Table 11.3. Based on the results in Table 11.3, both the cross-stitch, multi-task approach combined with the uncertainty-weighted loss led to the highest predictive performance for *frustration*. However, the highest performance for *engagement* came from the single-task random forest models. It appears that any underlying relationships between *engagement* and *frustration* that exists between these two affective states are primarily predictive of *frustration* but not *engagement*.

To evaluate the statistical significance of the multi-task cross-stitch model for *frustration*, the same non-parametric Wilcoxon signed rank test was performed based on the AUC for each of the 10 cross-validation folds, but the difference in AUC was determined to not be significant using

Table 11.3. Model evaluation results for single-task and multi-task models (*JavaTutor*).

Engagement				
Model Type	**Model**	**AUC**	**Accuracy**	**F1 Score**
Single-Task	**RF**	**0.539**	**0.551**	**0.272**
Multi-Task	RF	0.527	0.541	0.200
Frustration				
Model Type	**Model**	**AUC**	**Accuracy**	**F1 Score**
Single-Task	FFNN	0.680	0.785	0.168
Multi-Task	**CS (W)**	**0.694**	**0.880**	**0.245**

a p-value threshold of 0.05 (p=.084). As a result, additional experimentation is needed to evaluate the generalizability of this approach.

There are a number of factors to consider that varied between the *JavaTutor* and USMA corpora that may have had an impact on the performance of these techniques, such as the difference between predicting affect sequences from the same learning session vs. predicting multiple affective states concurrently. A result of this difference is that the use of the subsequent affective state as an auxiliary multi-task learning approach cannot be evaluated for generalizability using the *JavaTutor* data corpus. Additionally, the affect labels in this case were limited to one per session in the *JavaTutor* corpus, and thus the use of prior affective states as an input feature to boost predictive performance was not feasible in this case. As a result, the labels within the multi-task learning approach for the *JavaTutor* dataset represented a wide range of student behavioral cues over the span of up to an hour, while individual affect labels from the USMA dataset were only representative of the prior 20 seconds of student behavior. As a result, the low level of granularity within the *JavaTutor* features may not be informative enough for a MTL model focused on only two distinct affective states. Based on these results, we are unable to conclude that our demonstrated MTL pipeline is effectively generalizable across different datasets, and therefore more evaluations are needed across different learning tasks, affective states, and datasets where

affective sequence data could be used to perform additional experiments. This may necessitate future data collections where students self-report instances of concurrent affective states at multiple intervals throughout a single learning session, or additional data collections that implement third party field observations such as the BROMP protocol.

CHAPTER 12

CONCLUSION

Affect is a critical component in student learning. Automatically recognizing students' affective states is foundational to the development of user-adaptive learning environments that can support emotion regulation and promote enhanced learning experiences. Multimodal affect detectors have demonstrated the capability to effectively integrate sensor-free and sensor-based data channels, capturing multiple simultaneous perspectives on student interactions with digital learning environments. However, important questions remain about the predictive value of specific modalities, how they should be combined, and the optimal modeling techniques for multimodal data. Additionally, there is a need for continued research on how effectively multimodal affect detection techniques translate to alternative learning environments, educational subjects, and modalities.

However, multimodal affect detection systems can experience issues that impede the data capture process, such as mistracking, data storage constraints, software or hardware failure, and data transfer problems. Since the simple removal of missing or invalid samples can adversely impact the predictive performance of machine learning-based affect detection models induced from the data, more advanced techniques are needed. Therefore, increasing the amount of available multimodal data for training the affect models through synthetic data generation or augmentation in addition to imputation of the missing data values is a critical component of affective computing. As a result, it is also desirable to determine how well different augmentation and imputation techniques generalize to various modalities, student populations, and affect annotation techniques. This chapter provides a summary of the presented work and empirical results of the multimodal

affect detection framework, the outcomes of the hypotheses in Section 1.1, and avenues for future work for different components of the framework.

12.1 Hypotheses Revisited

This dissertation investigates the following hypotheses that address different components of the presented multimodal affect detection framework. The different hypotheses were primarily evaluated using the USMA dataset consisting of posture- and interaction-based modalities, and subsequently evaluated for generalizability using a different data corpus consisting of dialogue, interaction, and facial expression data.

- **Hypothesis 1:** Deep learning techniques achieve higher performance of student affect in terms of predictive accuracy and data augmentation impact when compared to standard non-deep learning classification models and upsampling techniques. Deep learning multi-task models trained using a student's affect sequence data outperform single-task deep learning models and non-deep learning models in terms of predictive accuracy.

    For data augmentation, the results indicate that the use of the auxiliary classifier generative adversarial network as a means generate synthetic training data led to improved predictive performance for the multimodal affect detection models, particularly when combined with the Wasserstein filtering to ensure similarity to the distributions of the original data. This deep learning approach outperformed multiple non-neural upsampling techniques when evaluated by affect classification performance. Additionally, the discriminator component of the data augmentation model often achieved higher predictive performance both of the neural and non-neural affect detection models.

    For multi-task learning, it was shown that the use of affect sequence data can be effectively used as both input features and auxiliary target variables to improve the

predictive performance of the multimodal affect models through the use of multi-task learning. In almost every case, the cross-stitch multi-task model significantly outperformed a series of non-neural single-task and multi-task models, in addition to neural multi-task models.

This hypothesis is accepted.

- **Hypothesis 2:** Deep learning imputation models such as conditional generative adversarial networks or stacked denoising autoencoders trained on multimodal data outperform standard non-deep learning imputation methods while having minimal adverse impact on student affect models and improving student affect model performance in terms of predictive accuracy.

  Stacked denoising autoencoders were shown in this work to be an effective means of improving multimodal affect detection in instances where a modality suffers from missing or invalid data. This approach outperformed other methods to addressing missing data, including removing incomplete multimodal feature vectors and non-neural imputation such as mean imputation. Deep generative models, including conditional variational autoencoders and conditional generative adversarial networks, were more frequently the optimal imputation technique across a series of evaluations with different masked modalities and artificial noise levels. The generative imputation approach achieved higher performance from two evaluation perspectives: raw imputation accuracy and impact on the predictive performance of multimodal affect models. Additionally, the deep generative imputation approach outperformed baseline imputation methods including mean imputation and probabilistic matrix factorization.

  This hypothesis is accepted.

- **Hypothesis 3:** Deep learning techniques that enhance multimodal affect detection are generalizable across two different learning environments, affect annotations, and modalities. These techniques are effective when applied within two distinct data corpora.

  The deep learning techniques addressing issues of data augmentation and imputation, in addition to improving predictive accuracy of affect detection, were evaluated using two distinct data corpora, consisting of different learning platforms, modalities, data representations, and affect annotation schematics. Empirical results indicate that multimodal approaches to affect detection offer improved performance compared to unimodal baselines. Additionally, the deep learning approaches to data augmentation as a means to improve predictive performance were shown to outperform non-neural baselines across both datasets. Multimodal autoencoders as a means of data imputation demonstrated improved performance over non-deep imputation approaches across both corpora as well. A majority of the evaluations confirm this hypothesis, however, some experiments did not result in the deep learning approaches achieving optimal performance. These experiments include the facial expression modality masking (non-neural imputation achieved the highest performance for engagement and frustration) and multi-task prediction of engagement (single-task random forest achieved the highest performance).

  This hypothesis is partially accepted.

12.2 Summary

This dissertation presents a multimodal framework that utilizes a variety of deep learning-based techniques to enhance the predictive capacities of student affect models. The preliminary evaluation of the core components of this framework was performed using a multimodal affect-centered dataset captured from students that were engaged with a game-based learning

environment for emergency medical care, *TC3Sim*. Using the EDA, posture, and gameplay trace logs from each student, the predictive performance of a number of machine learning models was investigated using a dataset consisting of the multimodal input data in addition to annotated instances of five distinct affective states using a field observation protocol, BROMP. During this process, three distinct variations of multimodal data fusion were also investigated to determine their impact on the affect models' predictive performance. These experiments were performed for two combinations of modalities: EDA + posture data, and posture + interaction data.

Additionally, a deep learning-based approach to synthetic data augmentation was explored using auxiliary classifier GANs. Combined with a filtering approach based on the Wasserstein distance metric, the described data augmentation approach induced higher predictive performance from different affect models through the generation of additional training data, outperforming several non-neural baseline approaches to data generation. The capability of the AC-GAN's discriminator component as a deep learning affect detection model was also investigated, with the results indicating that the use of the pre-trained discriminator was a promising alternative as a predictive model.

Finally, the preliminary work investigates several deep learning approaches to addressing missing or noisy data within modalities through multimodal data imputation. The first approach is the use of stacked denoising autoencoders, which was shown to effectively increase the predictive performance of different affect models through the enabling of a portion of the captured multimodal data that suffered from sensor failure during the initial data collection. Additionally, it was shown that the reconstruction of the imputed data to its original dimensionality aids both the interpretability of the imputed data and further improves the predictive performance of the affect models. The second approach utilizes two types of conditional generative models, C-GANs

and C-VAEs. The models were evaluated using artificial data masking for a sensor-based and interaction-based modality using varying levels of noise injection, with results indicating that the generative modeling approaches are more effective in terms of imputation accuracy as well as minimized impact on the pre-trained affect models' predictive performance compared to other non-neural baseline imputation approaches.

This dissertation also investigates the use of alternative neural architectures as a means to improve the predictive performance of the multimodal affect models. This includes the use of multi-task learning, which has been previously shown in other related work to achieve higher predictive performance compared to single-task models, while reducing the number of trained parameters required and the computational training time. Additionally, the use of cross-stitch models as a parameterization technique for the shared parameters between tasks is also investigated. The multi-task learning approach is implemented through the use of affect sequence data extracted from a student's learning session, and additionally, the use of a student's prior affective states are shown to further improve predictive performance across multiple affective states when used as input features in combination with the features from each modality. The dissertation also investigates the use of an uncertainty-weighted multi-task loss, which learns a different weighting for each task compared to a naïve uniform weighting commonly found in multi-task learning models. Results indicate that the use of multi-task learning for affect sequence modeling (particularly with cross-stitch networks) lead to statistically significant improvement across several different affective states, indicating that the inclusion of temporal contextual information as either independent or dependent variables is a means to improve multimodal affect detection performance.

Finally, this dissertation investigates the overall generalizability of this multimodal affect detection framework by evaluating each component using a different multimodal data corpus consisting of interaction trace data, student-tutor dialogue text, and facial expression data captured from undergraduate students that engage with a web-based development environment designed to teach introductory programming through a series of interactive coding exercises. This work also addresses generalizability across different affect annotation methods as the students self-reported levels of *frustration* and *engagement* at the end of each learning session, instead of having field observers annotate salient instances of different affective states in real time. Different alternative word embedding representations of the raw text are evaluated in terms of prediction performance of different multimodal affect models, and each of the deep learning approaches to data augmentation, imputation, and prediction are trained using this multimodal dataset. Overall results indicate that this multimodal affect detection framework leads to improved predictive performance, showing promise for future implementations. It is anticipated that the final outcome of the dissertation is a robust deep learning-based framework that is capable of reliably and accurately detecting and classifying student affect with a wide variety of captured modalities and learning environments. The presented framework shows promise for future integration into adaptive learning environments capable of dynamically conforming to an individual student's behavior and affective patterns, promoting emotion regulation, and providing an enhanced learning experience.

12.3 Future Work

The results of this dissertation provide several promising directions for future work. Investigating recent advances in multimodal machine learning techniques, including multimodal neural architectures, has strong potential to yield further improvements to the accuracy of multimodal

affect detectors in adaptive learning environments. This includes the use of sequential modeling such as Long Short-Term Memory (LSTM) networks or conditional random fields. Along similar lines, it would be worthwhile to evaluate the performance of the affect detection models using early prediction metrics to determine the time for each model to converge to a single prediction across a learning session. Additionally, further improvements can be made to the multimodal data processing pipeline, including more sophisticated approaches to the feature selection for each modality, data representation methods, feature engineering, and multi-task data upsampling. This includes the implementation and evaluation of the deep learning methods as regression-based models instead of classification models or using more granular discretization methods compared to a binary median split. Finally, the generalizability of the overall framework should continue to be empirically evaluated across different modality combinations and learning platforms, including extending the deep learning components of the framework to include domains outside of student modeling or affect detection.

Future work related to the data imputation component includes investigating more sophisticated denoising and data corruption techniques related to the framework's noise injection approach. Examining feature-level masking, data sample-level masking, or entirely missing modalities should also be considered. The presented multimodal data imputation approaches should also be investigated with additional sensor-based modalities that are commonly used in adaptive learning environments, such as facial expression, eye tracking, and physiological data, as these can all suffer from the same data collection issues mentioned in this work. One area of interest is the use of multiple modalities as the conditioning factors for each generative model, and conversely, the imputation of multiple modalities using a single conditioning modality. More complex neural architectures can also be evaluated as generative approaches to data imputation,

including auxiliary classifier generative adversarial networks (similar to Chapter 9), stacked variational autoencoders, and Wasserstein-based GANs (Arjovsky et al. 2017). Similar future work related to the generative modeling can also be applicable to the data augmentation work in Chapter 9, including the use of more sophisticated neural architectures for the generator and discriminator components of the AC-GAN, and the application of the presented methods across multiple modalities concurrently, or a single deep learning model for each individual modality (e.g., Early Fusion vs. Late Fusion).

With regard to multi-task learning and affect sequence modeling, the tradeoff between temporal information and model generalizability can be further explored by using different representations of the subsequent affective states instead of the one-hot encoding of a single state. Additionally, the generalizability of the affective dynamics-based multi-task modeling approach should be evaluated using different learning environments, student populations, and modalities, including an investigation of different methods of capturing sequential affect labels from a single learning session without compromising the intrusiveness of naturalism of the multimodal data collection, which could inadvertently lead to heightened levels of negatively-valenced affective states such as *frustration*. Our approach is dependent on the coding scheme used to annotate students' affective states in our dataset, so evaluating the performance of our models using different observational protocols or another method such as self-report would provide further insight into the impact that these factors have on our modeling approach.

Finally, it will be important to investigate the run-time integration of the proposed multimodal affect detectors. Additional analysis should be conducted on the feasibility and performance of the data imputation methods in a real time environment, and whether the imputation methods can maintain the predictive capabilities of the affect models in instances where

certain modalities suffer from issues with data capture. The run-time integration of the complete multimodal affect detection pipeline into different learning environments lays the foundation for enabling adaptive features such as user-sensitive feedback or tailored scaffolding to improve learner engagement and support greater learning outcomes.

REFERENCES

Akuzawa, K., Iwasawa, Y., & Matsuo, Y. (2019). Expressive speech synthesis via modeling expressions with variational autoencoder. *ArXiv:1804.02135*.

AlZoubi, O., AlMakhadmeh, B., Tawalbeh, S., Yassien, M., & Hmeidi, I. (2020). A deep learning approach for classifying emotions from physiological data. *Proceedings of the 11th International Conference on Information and Communication Systems*, 214–219.

AlZoubi, O., D'Mello, S., & Calvo, R. (2012). Detecting naturalistic expressions of nonbasic affect using physiological signals. *IEEE Transactions on Affective Computing*, *3*(3), 298–310.

Andres, J. M. A. L., Ocumpaugh, J., Baker, R., Slater, S., Paquette, L., Jiang, Y., Karumbaiah, S., Bosch, N., Munshi, A., Moore, A., & Biswas, G. (2019). Affect sequences and learning in Betty's Brain. *Proceedings of the 9th International Conference on Learning Analytics & Knowledge*, 383-390.

Arjovsky, M., Chintala, S., & Bottou, L. (2017). Wasserstein GAN. *arXiv preprint: 1701.07875.*

Arora, S., Ge, R., Liang, Y., Ma, T., & Zhang, Y. (2017). Generalization and equilibrium in generative adversarial nets (GANs). *Proceedings of the International Conference on Machine Learning*, 224–232.

Arroyo, I., Cooper, D., Burleson, W., Woolf, B., Muldner, K., & Christopherson, R. (2009). Emotion sensors go to school. *Proceedings of the 14th International Conference on Artificial Intelligence in Education*, 17–24.

Arulkumaran, K., Deisenroth, M., Brundage, M., & Bharath, A. (2017). Deep reinforcement learning: A brief survey. *IEEE Signal Processing Magazine*, *34*(6), 26–38.

Baker, R., D'Mello, S., Rodrigo, Ma. M., & Graesser, A. (2010). Better to be frustrated than bored: The incidence, persistence, and impact of learners' cognitive–affective states during interactions with three different computer-based learning environments. *International Journal of Human-Computer Studies*, *68*(4), 223–241.

Baker, R., Gowda, S., Wixon, M., Kalka, J., Wagner, A., Salvi, A., Aleven, V., Kusbit, G., Ocumpaugh, J., & Rossi, L. (2012). Towards sensor-free affect detection in cognitive tutor algebra. *Proceedings of the 5th International Conference on Educational Data Mining*, 126–133. International Educational Data Mining Society.

Baltrušaitis, T., Ahuja, C., & Morency, L. (2019). Multimodal machine learning: A survey and taxonomy. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *41*(2), 423–443.

Bosch, N., Chen, H., D'Mello, S., Baker, R., & Shute, V. (2015). Accuracy vs. Availability heuristic in multimodal affect detection in the wild. *Proceedings of the 17th International Conference on Multimodal Interaction*, 267–274. New York, NY, USA: Association for Computing Machinery.

Bosch, N., D'Mello, S., Baker, R., Ocumpaugh, J., & Shute, V. (2015). Temporal generalizability of face-based affect detection in noisy classroom environments. In C. Conati, N. Heffernan, A. Mitrovic, & M. F. Verdejo (Eds.), *Artificial Intelligence in Education* (pp. 44–53).

Bosch, N., D'Mello, S., Baker, R., Ocumpaugh, J., Shute, V., Ventura, M., & Zhao, W. (2016). Detecting student emotions in computer-enabled classrooms. *Proceedings of the International Joint Conference on Artificial Intelligence*, 4125–4129.

Botelho, A., Baker, R., & Heffernan, N. (2017). Improving sensor-free affect detection using deep learning. In E. André, R. Baker, X. Hu, Ma. M. T. Rodrigo, & B. du Boulay (Eds.), *Proceedings of the 18th International Conference on Artificial Intelligence in Education* (pp. 40–51).

Brave, S., & Nass, C. (2009). Emotion in human-computer interaction. In *The Human-Computer Interaction Handbook: Fundamentals, Evolving Technologies and Emerging Applications* (pp. 81–96). Hillsdale, NJ: Erlbaum Associates, Inc.

Cabada, R., Estrada, M., Hernández, F., & Bustillos, R. (2015). An affective learning environment for Java. *Proceedings of The 15th International Conference on Advanced Learning Technologies*, 350–354.

Calvo, R., & D'Mello, S. (2010). Affect detection: An interdisciplinary review of models, methods, and their applications. *IEEE Transactions on Affective Computing*, *1*(1), 18–37.

Carpenter, D., Emerson, A., Mott, B., Saleh, A., Glazewski, K., Hmelo-Silver, C., & Lester, J. (2020). Detecting off-task behavior from student dialogue in game-based collaborative learning. *Proceedings of the 21st International Conference on Artificial Intelligence in Education*, 56-66.

Caruana, R., & De Sa, V. (1996). Promoting poor features to supervisors: Some inputs work better as outputs. *Proceedings of the 9th International Conference on Neural Information Processing Systems*, 389-395.

Chan, M., Ochoa, X., & Clarke, D. (2020). *Multimodal learning analytics in a laboratory classroom* (M. Virvou, E. Alepis, G. Tsihrintzis, & L. Jain, Eds.). Cham: Springer International Publishing.

Chatziagapi, A., Paraskevopoulos, G., Sgouropoulos, D., Pantazopoulos, G., Nikandrou, M., Giannakopoulos, T., Katsamanis, A., Potamianos, A., & Narayanan, S. (2019). Data augmentation using GANs for speech emotion recognition. *Interspeech 2019*, 171–175. ISCA.

Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research*, *16*, 321–357.

Chen, M., Shi, X., Zhang, Y., Wu, D., & Guizani, M. (2017). Deep features learning for medical image analysis with convolutional autoencoder neural network. *IEEE Transactions on Big Data*, *1*(1), 1–10.

Chen, S., & Jin, Q. (2015). Multi-modal dimensional emotion recognition using recurrent neural networks. *Proceedings of the 5th International Workshop on Audio/Visual Emotion Challenge*, 49–56. Association for Computing Machinery.

Cohen, J. (1960). A coefficient of agreement for nominal scales. *Educational and Psychological Measurement*, *20*(1), 37–46.

Cooper, D. G., Arroyo, I., & Woolf, B. P. (2011). Actionable affective processing for automatic tutor interventions. In *New perspectives on affect and learning technologies* (pp. 127–140). New York, NY: Springer.

Craig, S., Graesser, A., Sullins, J., & Gholson, B. (2004). Affect and learning: An exploratory look into the role of affect in learning with AutoTutor. *Journal of Educational Media*, *29*(3), 241–250.

DeFalco, J., Rowe, J., Paquette, L., Georgoulas-Sherry, V., Brawner, K., Mott, B., Baker, R., & Lester, J. (2018). Detecting and addressing frustration in a serious game for military training. *International Journal of Artificial Intelligence in Education*, *28*(2), 152–193.

Deng, L., Hinton, G., & Kingsbury, B. (2013). New types of deep neural network learning for speech recognition and related applications: An overview. *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, 8599–8603.

Deng, L., & Platt, J. (2014). Ensemble deep learning for speech recognition. *Proceedings of the 15th Annual Conference of the International Speech Communication Association*, 1915–1919.

D'Mello, S. (2013). A selective meta-analysis on the relative incidence of discrete affective states during learning with technology. *Journal of Educational Psychology*, *105*(4), 1082–1099.

D'Mello, S., & Graesser, A. (2010). Mining bodily patterns of affective experience during learning. *Proceedings of The 3rd International Conference on Educational Data Mining*, 31–40.

D'Mello, S., & Graesser, A. (2011). The half-life of cognitive-affective states during complex learning. *Cognition and Emotion*, *25*(7), 1299–1308.

D'Mello, S., & Graesser, A. (2012). Dynamics of affective states during complex learning. *Learning and Instruction*, *22*(2), 145–157.

D'Mello, S., & Kory, J. (2012). Consistent but modest: A meta-analysis on unimodal and multimodal affect detection accuracies from 30 studies. *Proceedings of the 14th International Conference on Multimodal Interaction*, 31–38.

D'Mello, S., & Kory, J. (2015). A review and meta-analysis of multimodal affect detection systems. *ACM Computing Surveys*, *47*(3), 1–36.

D'Mello, S., Lehman, B., Pekrun, R., & Graesser, A. (2014). Confusion can be beneficial for learning. *Learning and Instruction*, *29*(1), 153–170.

Devlin, J., Chang, M., Lee, K., & Toutanova, K. (2018). *arXiv:1810.04805.*

Ekman, P., Friesen, W. V., & Hager, J. C. (2002). Facial Action Coding System. *A Human Face*.

Elkahky, A. M., Song, Y., & He, X. (2015). A multi-view deep learning approach for cross domain user modeling in recommendation systems. *Proceedings of the 24th International Conference on World Wide Web*, 278–288. Republic and Canton of Geneva, CHE.

Fung, P., Dey, A., Siddique, F. B., Lin, R., Yang, Y., Bertero, D., Wan, Y., Chan, R. H. Y., & Wu, C. S. (2016). Zara: A virtual interactive dialogue system incorporating emotion, sentiment, and personality recognition. *Proceedings of the International Conference on Computational Linguistics*, 278-281.

Geden, M., Emerson, A., Rowe, J., Azevedo, R., & Lester, J. (2020). Predictive student modeling in educational games with multi-task learning. *Proceedings of the AAAI Conference on Artificial Intelligence*, *34*, 654–661.

Geden, M., Emerson, A., Carpenter, D., Rowe, J., Azevedo, R., & Lester, J. (2021). Predictive student modeling in game-based learning environments with word embedding representations of reflection. *International Journal of Artificial Intelligence in Education, 31*(1), 1-23.

Ghaleb, E., Popa, M., & Asteriadis, S. (2019). Multimodal and temporal perception of audio-visual cues for emotion recognition. *Proceedings of the 8th International Conference on Affective Computing and Intelligent Interaction*, 552-558.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., & Bengio, Y. (2014). Generative adversarial networks. *ArXiv:1406.2661*.

Grafsgaard, J., Boyer, K., Wiebe, E., & Lester, J. (2012). Analyzing posture and affect in task-oriented tutoring. *Proceedings of the International Conference of the Florida Artificial Intelligence Research Society*, 438–443.

Grafsgaard, J., Wiggins, J., Boyer, K., Wiebe, E., & Lester, J. (2013a). Automatically recognizing facial expression: Predicting engagement and frustration. *Proceedings of The 6th International Conference on Educational Data Mining*, 43–50.

Grafsgaard, J., Wiggins, J., Boyer, K., Wiebe, E., & Lester, J. (2013b). Automatically recognizing facial indicators of frustration: A learning-centric analysis. *Proceedings of the 2013 Humane Association Conference on Affective Computing and Intelligent Interaction*, 159–165.

Grafsgaard, J., Wiggins, J., Vail, A., Boyer, K., Wiebe, E., & Lester, J. (2014). The additive value of multimodal features for predicting engagement, frustration, and learning during tutoring. *Proceedings of the 16th International Conference on Multimodal Interaction*, 42–49.

Harley, J., Bouchet, F., & Azevedo, R. (2013). Aligning and comparing data on emotions experienced during learning with MetaTutor. In H. C. Lane, K. Yacef, J. Mostow, & P. Pavlik (Eds.), *Proceedings of the 16th International Conference on Artificial Intelligence in Education* (pp. 61–70).

Harley, J., Bouchet, F., Hussain, M., Azevedo, R., & Calvo, R. (2015). A multi-componential analysis of emotions during complex learning with an intelligent multi-agent system. *Computers in Human Behavior*, *48*(1), 615–625.

Hart, S. G., & Staveland, L. E. (1988). Development of NASA-TLX (Task Load Index): Results of empirical and theoretical research. *Human Mental Workload*, 139-183.

He, L., Jiang, D., Yang, L., Pei, E., Wu, P., & Sahli, H. (2015). Multimodal affective dimension prediction using deep bidirectional long short-term memory recurrent neural networks. *Proceedings of the 5th International Workshop on Audio/Visual Emotion Challenge*, 73–80.

Henderson, N., Emerson, A., Rowe, J., & Lester, J. (2019). Improving sensor-based affect detection with multimodal data imputation. *2019 8th International Conference on Affective Computing and Intelligent Interaction (ACII)*, 669–675.

Henderson, N., Min, W., Rowe, J., & Lester, J. (2020a). Enhancing affect detection in game-based learning environments with multimodal conditional generative modeling. *Proceedings of the 2020 International Conference on Multimodal Interaction*, 134–143.

Henderson, N., Min, W., Rowe, J., & Lester, J. (2020b). Multimodal Player Affect Modeling with Auxiliary Classifier Generative Adversarial Networks. *Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment*, *16*(1), 224–230.

Henderson, N., Kumaran, V., Min, W., Mott, B., Wu, Z., Boulden, D., Lord. T., Reichsman, F., Dorsey, C., Wiebe, E., & Lester, J. (2020c). Enhancing student competency models for game-based learning with a hybrid stealth assessment framework. *Proceedings of the 13th International Conference on Educational Data Mining,* 92-103.

Henderson, N., Rowe, J., Mott, B., Brawner, K., Baker, R., & Lester, J. (2019). 4D affect detection: Improving frustration detection in game-based learning with posture-based temporal data fusion. *Artificial Intelligence in Education*, 144–156.

Henderson, N., Rowe, J., Paquette, L., Baker, R., & Lester, J. (2020). Improving affect detection in game-based learning with multimodal data fusion. In I. I. Bittencourt, M. Cukurova, K. Muldner, R. Luckin, & E. Millán (Eds.), *Proceedings of the 21st International Conference on Artificial Intelligence in Education* (pp. 228–239).

Henderson, N., Min, W., Emerson, A., Rowe, J., Lee, S., & Lester, J. (2021a). Early prediction of museum visitor engagement with multimodal adversarial domain adaptation. *Proceedings of the International Conference on Educational Data Mining Society*, 93-104.

Henderson, N., Min, W., Rowe, J., & Lester, J. (2021b). Enhancing multimodal affect recognition with multi-task affective dynamics modeling. *Proceedings of the 9th International Conference on Affective Computing and Intelligent Interaction,* 1-8.

Hinton, G., Osindero, S., & Teh, Y. (2006). A fast learning algorithm for deep belief nets. *Neural Computation*, *18*(7), 1527–1554.

Hinton, G., & Salakhutdinov, R. (2006). Reducing the dimensionality of data with neural networks. *Science*, *313*(5786), 504–507.

Hochreiter, S. (1998). The vanishing gradient problem during learning recurrent neural nets and problem solutions. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, *6*(2), 107–116.

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, *9*(8), 1735–1780.

Hsu, W., Zhang, Y., & Glass, J. (2017). Unsupervised domain adaptation for robust speech recognition via variational autoencoder-based data augmentation. *Proceedings of the IEEE Automatic Speech Recognition and Understanding Workshop*, 16–23.

Hutto, C., & Golbert, E. (2014). Vader: A parsimonious rule-based model for sentiment analysis of social media text. *Proceedings of the International AAAI Conference on Web and Social Media*, 216-225.

Isola, P., Zhu, J., Zhou, T., & Efros, A. (2017). Image-to-image translation with conditional adversarial networks. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 1125–1134.

Ivakhnenko, A., & Lapa, V. (1967). *Cybernetics and forecasting techniques*. Retrieved from https://cds.cern.ch/record/209675

Jaques, N., Taylor, S., Sano, A., & Picard, R. (2017). Multimodal autoencoder: A deep learning approach to filling in missing sensor data and enabling better mood prediction. *Proceedings of the Seventh International Conference on Affective Computing and Intelligent Interaction*, 202–208.

Jiang, Y., Bosch, N., Baker, R., Paquette, L., Ocumpaugh, J., Andres, J. Ma. A., Moore, A., & Biswas, G. (2018). Expert feature-engineering vs. deep neural networks: Which is better for sensor-free affect detection? *Proceedings of the International Conference on Artificial Intelligence in Education*, 198–211.

Kalimeri, K., & Saitis, C. (2016). Exploring multimodal biosignal features for stress detection during indoor mobility. *Proceedings of the 18th International Conference on Multimodal Interaction*, 53–60.

Karras, T., Aila, T., Laine, S., & Lehtinen, J. (2018). Progressive growing of GANs for improved quality, stability, and variation. *ArXiv:1710.10196*.

Kassam, K., & Mendes, W. (2013). The effects of measuring emotion: Physiological reactions to emotional situations depend on whether someone is asking. *PLOS ONE*, *8*(6), e64959.

Kendall, A., Gal, Y., Cipolla, R. (2018). Multi-task learning using uncertainty to weigh losses for scene geometry and semantics. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 7842-7491.

Kingma, D., & Ba, J. (2017). Adam: A method for stochastic optimization. *ArXiv:1412.6980*.

Kingma, D., & Welling, M. (2014). Auto-encoding variational bayes. *ArXiv:1312.6114*.

Komatani, K., & Okada, S. (2021). Multimodal human-agent dialogue corpus with annotations at utterance and dialogue levels. *Proceedings of the 9th International Conference on Affective Computing and Intelligent Interaction*, 1-8.

Kratzwald, B., Ilić, S., Kraus, M., Feuerriegel, S., & Prendinger, H. (2018). Deep learning for affective computing: Text-based emotion recognition in decision support. *Decision Support Systems*, *115*, 24–35.

Krokotsch, T., & Böck, R. (2019). Generative adversarial networks and simulated+unsupervised learning in affect recognition from speech. *Proceedings of the 8th International Conference on Affective Computing and Intelligent Interaction*, 28–34.

LeCun, Y., Boser, B., Denker, J., Henderson, D., Howard, R., Hubbard, W., & Jackel, L. (1989). Backpropagation applied to handwritten zip code recognition. *Neural Computation*, *1*(4), 541–551.

Lehman, B., D'Mello, S., & Graesser, A. (2012). Confusion and complex learning during interactions with computer learning environments. *The Internet and Higher Education*, *15*(3), 184–194.

Liu, M., & Tuzel, O. (2016). Coupled generative adversarial networks. *ArXiv:1606.07536*.

Liu, X., Gao, J., He, X, Deng, L., Duh, K., & Wang, Y. Y. (2015). Representation learning using multi-task deep neural networks for semantic classification and information retrieval. *Proceedings of the 2015 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, 912-921.

Littlewort, G., Whitehill, J., Wu, T., Fasel, I., Frank, M., Movellan, J. R., & Bartlett, M.S. (2011). The computer expression recognition toolbox (CERT). *Proceedings of the IEEE International Conference on Automatic Face and Gesture Recognition*, 298-305.

Loderer, K., Pekrun, R., & Lester, J. (2018). Beyond cold technology: A systematic review and meta-analysis on emotions in technology-based learning environments. *Learning and Instruction*, *70*, 101–162.

Long, M., Cao, Z., Wang, J., & Yu, P. (2017). Learning multiple tasks with multilinear relationship networks. *Proceedings of the 31st International Conference on Neural Information Processing Systems*, 1593-1602.

Ma, Y., Nguyen, K. L., Xing, F. Z., & Cambria, E. (2020). A survey on empathetic dialogue systems. *Information Fusion, 64*, 50-70.

McCulloch, W., & Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity. *The Bulletin of Mathematical Biophysics*, *5*(4), 115–133.

Meudt, S., Schmidt-Wack, M., Honold, F., Schüssel, F., Weber, M., Schwenker, F., & Palm, G. (2016). Going further in affective computing: How emotion recognition can improve adaptive user interaction. In *Intelligent Systems Reference Library*: *Vol. 1. Toward Robotic Socially Believable Behaving Systems* (pp. 73–103). Cham: Springer International Publishing.

Mierswa, I., Wurst, M., Klinkenberg, R., & Scholz, M. (2006). Yale: Rapid prototyping for complex data mining tasks. *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 935–940.

Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). Efficient estimation of word representations in vector space. *arXiv:1301.3781*.

Min, W., Mott, B., Rowe, J., Taylor, R., Wiebe, E., Boyer, K., & Lester, J. (2017). Multimodal goal recognition in open-world digital games. *Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment*, *13*.

Mirza, M., & Osindero, S. (2014). Conditional generative adversarial nets. *ArXiv:1411.1784*.

Misra, I., Shrivastava, A., Gupta, A., & Hebert, M. (2016). Cross-stitch networks for multi-task learning. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 3994–4003.

Mitchell, C., Ha, E., Boyer, K., & Lester, J. (2013). Learner characteristics and dialogue: Recognizing effective and student-adaptive tutorial strategies. *International Journal of Learning Technology, 8*(4), 382-403.

O'Brien, H. L., & Toms, E. G. (2010). The development and evaluation of a survey to measure user engagement. *Journal of the American Society for Information Science and Technology, 61*(1), 50-69.

Ocumpaugh, J., Baker, R., & Rodrigo, M. M. (2015). *Baker Rodrigo Ocumpaugh Monitoring Protocol (BROMP) 2.0 Technical and Training Manual*. 1–72.

Odena, A., Olah, C., & Shlens, J. (2017). Conditional image synthesis with auxiliary classifier GANs. *Proceedings of the International Conference on Machine Learning*, 2642–2651. PMLR.

Pardos, Z., Baker, R., San Pedro, M., Gowda, S., & Gowda, S. (2014). Affective states and state tests: Investigating how affect and engagement during the school year predict end-of-year learning outcomes. *Journal of Learning Analytics*, *1*(1), 107–128.

Park, K., Mott, B., Min, W., Boyer, K., Wiebe, E., & Lester, J. (2019). Generating educational game levels with multistep deep convolutional generative adversarial networks. *Proceedings of the IEEE Conference on Games*, 1–8.

Pekrun, R., Goetz, T., Daniels, L., Stupnisky, R., & Perry, R. (2010). Boredom in achievement settings: Exploring control–value antecedents and performance outcomes of a neglected emotion. *Journal of Educational Psychology*, *102*(3), 531–549.

Pennebaker, J., Boyd, R., Jordan, K., & Blackburn, K. (2015). The development and psychometric properties of LIWC2015. University of Texas at Austin.

Pennington, J., Socher, R., & Manning, C. (2014). Glove: Global vectors for word representation. *Proceedings of the 2014 Conference on Emperical Methods in Natural Language Processing*, 1532-1543.

Peters, M., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., & Zettlemoyer, L. (2018). Deep contextualized word representations. *Proceedings of the 2018 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, 2227-2237.

Picard, R. (1999). Affective Computing for HCI. *Proceedings of The 8th International Conference on Human-Computer Interaction: Ergonomics and User Interfaces*, *1*, 829–833. USA: L. Erlbaum Associates Inc.

Picard, R. (2000). *Affective Computing*. MIT Press.

Picard, R. (2010). Affective computing: From laughter to IEEE. *IEEE Transactions on Affective Computing*, *1*(1), 11–17.

Platt, J. (1998). *Sequential minimal optimization: A fast algorithm for training support vector machines*.

Poole, B., Sohl-Dickstein, J., & Ganguli, S. (2014). Analyzing noise in autoencoders and deep networks. *ArXiv:1406.1831*.

Poria, S., Cambria, E., Bajpai, R., & Hussain, A. (2017). A review of affective computing: From unimodal analysis to multimodal fusion. *Information Fusion*, *37*, 98–125.

Psaltis, A., Kaza, K., Stefanidis, K., Thermos, S., Apostolakis, K., Dimitropoulos, K., & Daras, P. (2016). Multimodal affective state recognition in serious games applications. *Proceedings of The International Conference on Imaging Systems and Techniques*, 435–439.

Qiu, J., & Zhao, W. (2018). Data encoding visualization based cognitive emotion recognition with AC-GAN applied for denoising. *Proceedings of The 17th International Conference on Cognitive Informatics Cognitive Computing*, 222–227.

Rajendran, R., Kumar, A., Carter, K., Levin, D., & Biswas, G. (2018). Predicting learning by analyzing eye-gaze data of reading behavior. *Proceedings of The 11th International Conference on Educational Data Mining*, 455–461.

Ramachandran, B., Romero Pinto, S., Born, J., Winkler, S., & Ratnam, R. (2017). Measuring neural, physiological and behavioral effects of frustration. *Proceedings of The 17th International Conference on Biomedical Engineering*, 43–46.

Rodrigo, Ma. M. T., Baker, R., Jadud, M., Amarra, A., Dy, T., Espejo-Lahoz, M., Lim, S. A., Pascua, S., Sugay, J., & Tabanao, E. (2009). Affective and behavioral predictors of novice programmer achievement. *Proceedings of The 14th Annual ACM SIGCSE Conference on Innovation and Technology in Computer Science Education*, 156–160.

Rumelhart, D., Hinton, G., & Williams, R. (1986). In *Parallel Distibuted Processing: Explorations in the Microstructure of Cognition*: *Vol. 1*. *Learning internal representations by error propagation* (pp. 318–362).

Sabourin, J., Rowe, J., Mott, B., & Lester, J. (2013). Considering alternate futures to classify off-task behavior as emotion self-regulation: A supervised learning approach. *Journal of Educational Data Mining*, *5*(1), 9–38.

Salakhutdinov, R., & Mnih, A. (2008). Probabilistic matrix factorization. *Advances in Neural Information Processing Systems*, 1257–1264.

Sanghvi, J., Castellano, G., Leite, I., Pereira, A., McOwan, P. W., & Paiva, A. (2011). Automatic analysis of affective postures and body motion to detect engagement with a game companion. *Proceedings of the 6th International Conference on Human-Robot Interaction*, (March), 305–312. ACM.

Sawyer, R., Smith, A., Rowe, J., Azevedo, R., & Lester, J. (2017). Enhancing student models in game-based learning with facial expression recognition. *Proceedings of the 25th Conference on User Modeling, Adaptation and Personalization*, 192–201.

Sennrich, R., Haddow, B., & Birch, A. (2016). Neural machine translation of rare words with subword units. *ArXiv:1508.07909*.

Shui, C., Abbasi, M., Robitaille, L.E., Wang, B., & Gagne, C. (2019). A principled approach for learning task similarity in multi-task learning. *Proceedings of the 28th International Joint Conference on Artificial Intelligence*, 3446-3452.

Sohn, K., Yan, X., & Lee, H. (2015). Learning structured output representation using deep conditional generative models. *Advances in Neural Information Processing Systems*, 3483–3491.

Soleymani, M., Asghari-Esfeden, S., Fu, Y., & Pantic, M. (2016). Analysis of EEG signals and facial expressions for continuous emotion detection. *IEEE Transactions on Affective Computing*, *7*(1), 17–28.

Soleymani, M., Pantic, M., & Pun, T. (2012). Multimodal emotion recognition in response to videos. *IEEE Transactions on Affective Computing*, *3*(2), 211–223.

Song, M., Yang, Z., Baird, A., Parada-Cabaleira, E., Zhang, Z., Zhao, Z., & Schuller, B. (2019). Audiovisual analysis for recognizing frustration during gameplay: Introducing the multimodal game frustration database. *Proceedings of the 8th International Conference on Affective Computing and Intelligent Interaction*, 517-523.

Sottilare, R., Baker, R., Graesser, A., & Lester, J. (2018). Special issue on the Generalized Intelligent Framework for Tutoring (GIFT): Creating a stable and flexible platform for innovations in AIED research. *International Journal of Artificial Intelligence in Education*, *28*(2), 139–151.

Spain, R., Rowe, J., Goldberg, B., Pokorny, R., & Lester., J. (2019). Enhancing learning outcomes through adaptive remediation with GIFT. *Proceedings of the Interservice/Industry Training, Simulation and Education Conference*, 1–11.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, *15*(56), 1929–1958.

Sun, J., Huang, M. X., Ngai, G., & Chan, S. C. F. (2014). *Non-intrusive multimodal attention detection* (PhD Thesis).

Tao, J., & Tan, T. (2005). Affective computing: A review. *Proceedings of The International Conference on Affective Computing and Intelligent Interaction*, 981–995.

Thomas, C., Nair, N., & Jayagopi, D. (2018). Predicting engagement intensity in the wild using temporal convolutional network. *Proceedings of the 20th International Conference on Multimodal Interaction*, 604–610.

Tiam-Lee, T., & Sumi, K. (2018). Adaptive feedback based on student emotion in a system for programming practice. *Proceedings of the International Conference on Intelligent Tutoring Systems*, 243–255.

Tu, J., Liu, H., Meng, F., Liu, M., & Ding, R. (2018). Spatial-temporal data augmentation based on LSTM autoencoder network for skeleton-based human action recognition. *Proceedings of The 25th IEEE International Conference on Image Processing*, 3478–3482.

Tu, Z., Liu, Y., Shang, L., Liu, X., & Li, H. (2017). Neural machine translation with reconstruction. *Proceedings of the AAAI Conference on Artificial Intelligence*, *31*(1).

Tzeng, E., Hoffman, J., Saenko, K., & Darrell, T. (2017). Adversarial discriminative domain adaptation. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 7167–7176.

Vail, A., Grafsgaard, J., Boyer, K., Wiebe, E., & Lester, J. (2016). Predicting learning from student affective response to tutor questions. In A. Micarelli, J. Stamper, & K. Panourgia (Eds.), *Intelligent Tutoring Systems* (pp. 154–164). Cham: Springer International Publishing.

Vallender, S. (1974). Calculation of the Wasserstein distance between probability distributions on the line. *Theory of Probability & Its Applications*, *18*(4), 784–786.

Van Buuren, S., & Groothuis-Oudshoorn, K. (2011). MICE: Multivariate imputation by chained equations in R. *Journal of Statistical Software, 45,* 1-67.

VanLehn, K., Siler, S., Murray, C., Yamauchi, T., & Baggett, W. (2003). Why do only some events cause learning during human tutoring? *Cognition and Instruction*, *21*(3), 209–249.

Vincent, P., Larochelle, H., Lajoie, I., Bengio, Y., & Manzagol, P.-A. (2010). Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. *Journal of Machine Learning Research*, *11*(3), 3371–3408.

Wei, W., Jia, Q., & Chen, G. (2016). Real-time facial expression recognition for affective computing based on Kinect. *Proceedings of The 11th Conference on Industrial Electronics and Applications*, 161–165.

Weiss, T. G., Carayannis, T., Jolly, R., Weiss, T. G., Ca, T., & Jolly, R. (2016). Missing data: A systematic review of how they are reported and handled. *Epidemiology*, *12*(5), 729–732.

Werbos, P. (1974). Beyond Regression. *Ph.D. Dissertation, Harvard University*.

Whitehill, J., Serpell, Z., Lin, Y., Foster, A., & Movellan, J. (2014). The faces of engagement: Automatic recognition of student engagement from facial expressions. *IEEE Transactions on Affective Computing*, *5*(1), 86–98.

Wiggins, J., Grafsgaard, J., Boyer, K., Wiebe, E., & Lester, J. (2017). Do you think you can? The influence of student self-efficacy on the effectiveness of tutorial dialogue for computer science. *International Journal of Artificial Intelligence in Education, 27*(1), 130-153.

Worsley, M., Scherer, S., Morency, L. P., & Blikstein, P. (2015). Exploring behavior representation for learning analytics. In *Proceedings of the 2015 ACM on International Conference on Multimodal Interaction* (pp. 251-258).

Wu, S., Du, Z., Li, W., Huang, D., & Wang, Y. (2019). Continuous emotion recognition in videos by fusing facial expression, head pose and eye gaze. *Proceedings of the 21st International Conference on Multimodal Interaction*, 40–48.

Xu, W., Sun, H., Deng, C., & Tan, Y. (2017). Variational autoencoder for semi-supervised text classification. *Proceedings of the AAAI Conference on Artificial Intelligence*, *31*(1).

Yang, J., Wang, K., Peng, X., & Qiao, Y. (2018). Deep recurrent multi-instance learning with spatio-temporal features for engagement intensity prediction. *Proceedings of The 20th International Conference on Multimodal Interaction*, 594–598.

Yoon, J., Jordon, J., & Schaar, M. (2018). GAIN: Missing data imputation using generative adversarial nets. *Proceedings of The International Conference on Machine Learning*, 5689–5698. PMLR.

You, Q., Luo, J., Jin, H., & Yang, J. (2015). Joint visual-textual sentiment analysis with deep neural networks. *Proceedings of the 23rd ACM International Conference on Multimedia*, 1071–1074.

Young, T., Hazarika, D., Poria, S., & Cambria, E. (2018). Recent Trends in Deep Learning Based Natural Language Processing. *IEEE Computational Intelligence Magazine*, *13*(3), 55–75.

Zeiler, M. D. (2012). *ADADELTA: An adaptive learning rate method*. http://doi.acm.org.ezproxy.lib.ucf.edu/10.1145/1830483.1830503

Zeng, Z., Pantic, M., Roisman, G., & Huang, T. (2009). A survey of affect recognition methods: Audio, visual, and spontaneous expressions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *31*(1), 39–58.

Zhang, H., Xu, T., Li, H., Zhang, S., Wang, X., Huang, X., & Metaxas, D. (2017). StackGAN: Text to photo-realistic image synthesis with stacked generative adversarial networks. *Proceedings of The IEEE International Conference on Computer Vision*, 5907–5915.

Zhang, Y., & Yang, Q. (2017). Learning sparse task relations in multi-task learning. *Proceedings of the 31st AAAI Conference on Artificial Intelligence*, 2914-2920.

Zhao, Z., Zheng, P., Xu, S., & Wu, X. (2019). Object detection with deep learning: A review. *IEEE Transactions on Neural Networks and Learning Systems*, *30*(11), 3212–3232.

Zhong, Z., Li, J., Luo, Z., & Chapman, M. (2018). Spectral–spatial residual network for hyperspectral image classification: A 3-D deep learning framework. *IEEE Transactions on Geoscience and Remote Sensing*, *56*(2), 847–858.

Zhou, C., & Paffenroth, R. (2017). Anomaly detection with robust deep autoencoders. *Proceedings of The 23rd International Conference on Knowledge Discovery and Data Mining*, 665–674.

Zhu, X., Liu, Y., Li, J., Wan, T., & Qin, Z. (2018). Emotion classification with data augmentation using generative adversarial networks. *Advances in Knowledge Discovery and Data Mining*, 349–360.