Systematic Evaluation of Personalized Deep Learning Models for Affect Recognition

YUNJO HAN, KAIST, Republic of Korea
PANYU ZHANG, KAIST, Republic of Korea
MINSEO PARK, Sungkyunkwan University, Republic of Korea
UICHIN LEE*, KAIST, Republic of Korea

Understanding human affective states such as emotion and stress is crucial for both practical applications and theoretical research, driving advancements in the field of affective computing. While traditional approaches often rely on generalized models trained on aggregated data, recent studies highlight the importance of personalized models that account for individual differences in affective responses. However, there remains a significant gap in research regarding the comparative evaluation of various personalization techniques across multiple datasets. In this study, we address this gap by systematically evaluating widely-used deep learning-based personalization techniques for affect recognition across five open datasets (i.e., AMIGOS, ASCERTAIN, WESAD, CASE, and K-EmoCon). Our analysis focuses on realistic scenarios where models must adapt to new, unseen users with limited available data, reflecting real-world conditions. We emphasize the principles of reproducibility by utilizing open datasets and making our evaluation models and codebase publicly available. Our findings provide critical insights into the generalizability of personalization techniques, the data requirements for effective personalization, and the relative performance of different approaches. This work offers valuable contributions to the development of personalized affect recognition systems, fostering advancements in both methodology and practical application.

CCS Concepts: • **Human-centered computing** → **Ubiquitous and mobile computing**; • **Applied computing** → **Life and medical sciences**.

Additional Key Words and Phrases: Affective Computing, Deep Learning, Personalization, Open Datasets, Reproducibility

ACM Reference Format:
Yunjo Han, Panyu Zhang, Minseo Park, and Uichin Lee. 2024. Systematic Evaluation of Personalized Deep Learning Models for Affect Recognition. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 8, 4, Article 206 (December 2024), 35 pages. https://doi.org/10.1145/3699724

1 Introduction

The need to understand human affective states holds immense practical and theoretical value, as it contributes to and influences decision making and behavior changes [17, 56]. This necessity drives the field of *affective computing*, which aims to enable computers to recognize and understand human affect for intelligent and personalized interactions [60]. Furthermore, affective computing enables building persuasive technologies for promoting personal well-being and helping fulfill human potentialities [38]. Affect, as a multifaceted phenomenon, offers a range of detection cues, yet there is an increasing emphasis on recognizing it through physiological and

[*] Corresponding author

Authors' Contact Information: Yunjo Han, yjhan99@kaist.ac.kr, KAIST, Daejeon, Republic of Korea; Panyu Zhang, panyu@kaist.ac.kr, KAIST, Daejeon, Republic of Korea; Minseo Park, tim0726@g.skku.edu, Sungkyunkwan University, Suwon, Republic of Korea; Uichin Lee, uclee@kaist.edu, KAIST, Daejeon, Republic of Korea.

ACM 2474-9567/2024/12-ART206
https://doi.org/10.1145/3699724

behavioral signals. Physiological signals can reflect spontaneous affective responses beyond personal control, offering reliable affective state indicators [31]. Behavioral signals such as data from accelerometers offer insights into affective states by capturing unique movement patterns and user behaviors [63]. Utilizing these signals, researchers have developed machine learning (ML) models for affect recognition, yielding results that produce highly persuasive outcomes [15]. In recent years, deep learning (DL) has dominated ML research, motivating the development of DL-based affect recognition models [14, 82]. Rather than extracting features based on fixed rules, DL allows models to automatically learn complex information from raw physiological and behavioral sensor data, enabling the end-to-end approach [14].

To advance affect recognition models, researchers have identified that considering individual differences in affective responses is one key strategy for boosting model performance [9, 41, 74]. This is based on the fact that an identical stimulus may trigger varied affective responses among different individuals [5, 22]. Thus, generalized one-size-fits-all models, which use the aggregated data from all individuals indiscriminately for model training, might fail to address the distinct characteristics of each individual [46, 74]. Prior studies [9, 29, 87, 88] have presented various techniques for developing personalized models. Widely-used techniques encompass *user-specific modeling* with only a target user's data used for personalization, *hybrid modeling* with part of a target user's data aggregated into the overall training data, and *fine-tuning of general models* with part of a target user's data. There are also group-based personalization approaches, such as *cluster-specific modeling*, which groups similar users and creates models for each group, and *multi-task learning*, which involves tailoring the learning process to each group while simultaneously sharing information among them. These techniques have been demonstrated to enhance recognition performance across various applications [9, 29, 87, 88].

Yet, there exists a notable gap in research, as very few studies have concurrently compared the effectiveness of these diverse personalization techniques using multiple datasets. A comprehensive analysis is needed to understand the differences among various *personalized models* and to determine whether they truly outperform the *generalized models*. When evaluating such personalization techniques, it is important to iteratively hold out an individual from the dataset as an *unseen* test user, while the others' data are used as a training data set as in leave-one-participant-out (LOPO) cross-validation. This contrasts with many prior works that have trained and tested their models [9, 29, 74, 86] where the training data includes samples from every user, such that the test set users are also *seen* in the training set. Our focus is on model testing for these *new* users who have limited data available for model training, a scenario that mirrors real-world applications.

Moreover, the data analysis and model evaluations in previous studies often focused on only one or two datasets (see Table 1 and Table 2). Datasets were usually unpublished, as was the analysis code [45], and in some cases, there was a lack of detailed method descriptions. Evaluating each technique using multiple *open datasets* allows us to gain valuable insights into how personalization techniques work in diverse contexts. It is beneficial to the research community to openly share the evaluation process for further studies. This need reflects the growing focus on ensuring *reproducibility* to solidify the reliability and validity of ML research findings [3, 21, 45]. It necessitates not only the release of code and datasets with comprehensive details (*technical reproducibility*) but also the evaluation of models under new conditions, such as different datasets (*conceptual reproducibility*).

In this context, we take a step towards systematically evaluating well-known DL-based personalization techniques in affect recognition. To ensure reproducibility, we focus on open datasets gathered in controlled environments, rich in wearable physiological and behavioral signal data with user profile information (e.g., gender or personality traits). We apply a uniform data preprocessing pipeline across five different datasets, preparing them for input into diverse deep learning models, thus facilitating end-to-end learning. We then build well-known non-personalized (i.e., one-size-fits-all) and personalized affect recognition models proposed in prior studies. Subsequently, we compare their performance against each other, providing a comprehensive evaluation of the efficacy of each personalization technique across five datasets. Additionally, we have consolidated our

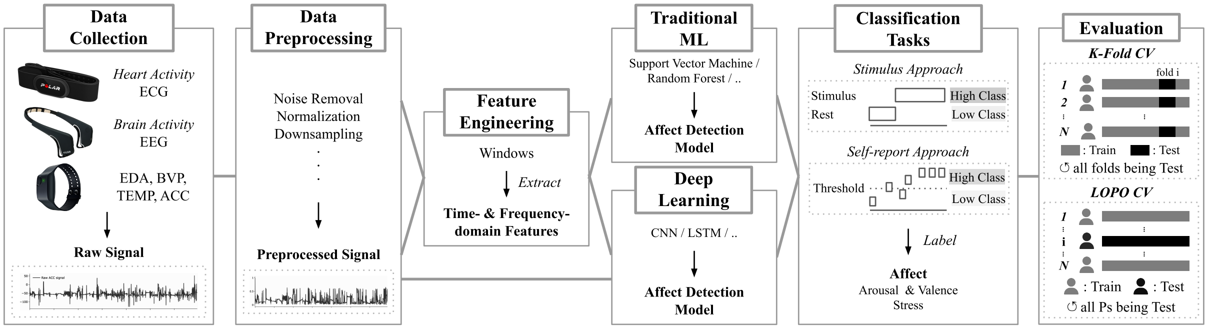


Fig. 1. Overview of General Affect Recognition Modeling and Evaluation Process.

implementation of all methods and made them publicly available, promoting the exploration and refinement of existing personalization strategies for further research.[1]

In this work, we present a set of contributions that advance the field of affect recognition by addressing key challenges in personalization and reproducibility as follows:

- *Systematic and Comprehensive Evaluation:* We systematically evaluated personalization techniques across multiple open datasets, offering a comprehensive comparison and revealing their consistency or lack thereof. Our study provides valuable insights for advancing these techniques and understanding their generalizability.
- *Emphasis on Reproducibility:* Our work emphasizes reproducibility by using open datasets and releasing our evaluation models and codebase. This transparency is crucial for validating and extending personalized models in ubiquitous computing.
- *Realistic Personalization Scenarios:* Unlike previous studies, we evaluate models on entirely unseen users, providing practical insights into how personalization strategies perform in real-world conditions, crucial for building affect recognition systems.
- *Data Requirements for User-dependent Personalization:* While it is acknowledged that personalization requires data from the unseen user, our study provides detailed insights into the specific data requirements about the amount necessary for effective personalization across different datasets.
- *Novelty in Aggregation and Comparative Analysis:* Our novelty lies in aggregating models and procedures for a unified, systematic comparative analysis, offering a holistic view of personalization techniques not previously presented.

2 Background and Related Works

2.1 Overview of Sensor-based Affect Recognition and Evaluation

With the advancement of sensor technology, recent research in affect recognition has actively utilized physiological and behavioral signals for affect detection. Studies attach wearable sensors to users to collect signal data and create detection models using ML algorithms after conducting ground truth labeling based on experimental protocols or self-reports [15, 19]. Figure 1 shows an overview of affect recognition modeling and evaluation.

Commonly considered signals include electroencephalography (EEG) and magnetoencephalography (MEG) related to brain activity, electrocardiography (ECG) and blood volume pressure (BVP) related to heart activity,

[1] https://github.com/Kaist-ICLab/Personalized_Affective_Computing.git

electromyogram (EMG) related to muscle activity, electrodermal activity (EDA), respiration (RESP), skin temperature (TEMP), and accelerometer (ACC). Researchers often use multiple signals simultaneously to build detection models. The typical affects targeted for detection are based on the dimensional affect model, classifying arousal and valence states as defined in Russell's circumplex model of affect [64]. In addition, we can model discrete emotions, such as stress, anger, and happiness, as defined in existing affect models [15]. In this work, we focus on binary classifications of arousal, valence, and stress levels as representative affect modeling. Russell et al. defined arousal as the intensity of emotions and valence as the intrinsic attractiveness (positive) or averseness (negative) of an emotional experience [64]. Although there is a lack of a universally accepted definition of stress, stress is often modeled as a dynamic process reflecting our physiological and bodily responses to emotional and physical stressors [53].

Two main labeling approaches have been introduced to obtain ground truth data for these states [79]. The stimulus approach involves labeling based on the presence or type of stimuli for emotion elicitation in controlled settings. Researchers use various stimuli, such as showing videos and pictures or administrating well-known stress-inducing tasks like the Trier social stress test (TSST). Affect labels are then assigned based on the elicited emotions or the presence of a stress task [15, 53]. Alternatively, the self-report approach uses affect annotations from study participants. Affect annotations can be labeled per event or continuously by retrospectively reviewing recorded events [58, 69]. Likert scale questionnaires have been frequently used, later classified into high or low affective states based on a specific threshold manually.

Subsequently, detection models are built and evaluated. Before signals are input into these models, they undergo preprocessing steps such as noise removal, normalization, and downsampling [14, 71]. Then, time- and frequency-domain features are calculated within a specified window, or automatic feature learning is performed using autoencoders [53, 71]. Extracted features are fed into traditional ML algorithms such as support vector machine (SVM), random forest (RF), or DL models such as convolutional neural networks (CNNs), long-short-term-memory recurrent neural networks (LSTM) to classify label values. Recently, the use of DL models for end-to-end learning has increased, and a growing number of studies show their potential [14, 39]. Preprocessed signals are segmented and directly fed into the model, reducing the complex process of feature engineering [43, 71].

To evaluate the performance of the built models, the cross-validation (CV) method is primarily used, allowing for the evaluation of the model's generalizability using data not seen during the learning process. This process includes K-fold CV, where each participant's dataset is divided into K equal segments, and each fold is used as a test set once while the others are used for training. It also involves leave-one-participant-out (LOPO) CV, where the model is trained on data from all participants except one, which is then used for testing, and this cycle is repeated for each individual participant. Lastly, metrics used for evaluation include classification accuracy, f1-score, area under the receiver operating characteristic (AUROC), and mean absolute error (MAE) [79].

2.2 Personalized Affect Recognition Modeling

Personalization in machine learning refers to creating models targeted towards specific individuals by understanding their unique characteristics [68]. In the field of affect recognition, various personalization methods have been used to overcome the inability of generalized models to account for individual differences, showing they can help recognize an individual's affect [41, 74]. Personalization methods used in previous studies can broadly be classified into **data-level** and **model-level** approaches. **Data-level** personalization is performed before inputting data into the model or classifier, whereas **Model-level** involves making changes to the model itself [41].

Data-level techniques include *user-specific* modeling, which involves creating separate models for each individual using only their data, and *cluster-specific* modeling, which involves creating separate models for groups classified based on certain criteria. Table 1 summarizes previous works utilizing data-level personalization techniques. For example, Zenonos et al. [88] conducted user-specific modeling for four users using only their

Table 1. Summary of Previous Works on Personalized Affect Recognition: Data-level. *Dataset* column contains detailed information about the datasets used for the evaluation. It indicates, in order, whether the datasets were self-collected by authors or are open datasets, whether they were collected in a controlled environment or in the wild, the type of data authors included (P: physiological, B: behavioral, C: Contextual), and the affect labels for detection.

Technique	Ref.	Method	Dataset	Evaluation	Result
User-specific	[88]	Only include individual's data for model training • Traditional model	• Self/Closed • In-the-wild • P, B • Mood	Compare personalized (Leave-one-out CV) and generalized (LOPO CV) model	Personalization increased accuracy
Cluster-specific	[2]	Only include the k-nearest neighbors to target participant's mobile sensing behavioral features for model training • Traditional model	• Open data [80], [81] • In-the-wild • B, C • Mood, Stress	Compare including only k neighbors (5,10,50,100,500) and all neighbors using LOPO CV	Including more neighbors generally decreased MAE
	[29]	Only include participants with same gender for model training • DL model	• Self/Closed • In-the-wild • C • PHQ	Compare personalized and generalized model using 3-Fold CV (for each participant, one hold-out fold as a test, 80% data of the remaining two folds as train, and 20% of the remaining as a validation set)	For daily prediction, personalization had little impact on MAE, similar to one-day-ahead forecasting
Both	[9]	Only include participants within same cluster based on PSS-14 for model training • Both Traditional and DL model	• Self/Closed • Controlled • P • Stress	Compare user-specific, cluster-specific, and generalized model (10% data as test set)	User-specific model's accuracy was the highest, followed by cluster-specific and generalized
	[77]	Only include participants within the same cluster based on a mean and standard deviation of each feature value • Traditional model	• Open data [67], Self/Closed • Controlled, In-the-wild • P, B, C • Stress	For [67], compare user-specific (20% data as test set), cluster-specific (LOPO CV), and generalized (LOPO CV) model For Self/Closed, compare user-specific (Leave-one-day-out), cluster-specific (LOPO CV), and generalized (LOPO CV) model	For [67], user-specific model increased accuracy and f1 score, but cluster-specific didn't For self-collected, personalization did not increase accuracy and f1 score
Others	[39]	While using autoencoder for automatic feature extraction, fine-tune the encoder using each individual's data	• Self/Closed • In-the-wild • P, B • Mood, health, stress	Compare personalized and generalized model in user-dependent setting (for each participant, 60% as train, 20% as validation, and 20% as test set)	Fine-tuning the autoencoder did not have impact on MAE

past physiological data and compared the performance with a generalized model that uses all other users' data, showing an average increase of about 8% in mood recognition accuracy. Can et al. [9] compared the performance of user-specific, cluster-specific (based on clustering using perceived stress scale scores), and generalized approaches for ambulatory stress detection using heart rate (HR) and EDA data. They found that user-specific, cluster-specific, and generalized approaches had decreasing levels of accuracy, respectively, suggesting that cluster-specific modeling could be an effective approach when user-specific data is insufficient. Tervonen et al. [77] also compared stress detection accuracy using user-specific, cluster-specific (based on clustering using individual averages and variances of extracted features), and generalized models, finding that user-specific models performed better than generalized ones, while cluster-specific models showed similar performance.

Model-level techniques include a *fine-tuning* approach and *multi-task learning* (*MTL*) approach. Table 2 summarizes previous works utilizing model-level personalization techniques.

Fine-tuning, a transfer learning technique, involves initializing a target network with a base network trained on a large amount of base data and then tuning it with a smaller amount of target data [85]. In affect recognition

Table 2. Summary of Previous Works on Personalized Affect Recognition: Model-level.

Technique	Ref.	Method	Dataset	Evaluation	Result
Fine-Tuning	[29]	First trains a baseline model using all participant data, then fine-tunes the last fully-connected and output layers for each individual • DL model	• Self/Closed • In-the-wild • C • PHQ	Compare personalized and generalized model using 3-Fold CV (for each participant, one hold-out fold as test, 80% data of the remaining two folds as train, and 20% of the remaining as validation set)	For daily prediction personalization had little impact on MAE, but for one-day-ahead forecasting, personalization decreased MAE
	[7]	First trains a baseline model using all participant data, then fine-tunes all layers using a small amount of participant's data • DL model	• Open data [67], [34] • Controlled • P • Stress	Compare using personalized and generalized model using LOPO CV (use different portions, 1, 5, and 10% of test participants for tuning)	For [67], accuracy and f1 score increased with tuning above 5% For [34], accuracy and f1 score increased with tuning above 1%
	[87]	First trains a baseline model using all participant data, then fine-tunes all layers or last LSTM layer and output layers for each individual • DL model	• Self/Closed • In-the-wild • P, B, C • Mood, health, stress	Compare personalized and generalized model (use 80% of users for training, 20% for testing)	MAE decreased as portion sizes increases Using more than 30% for tuning all and 10% for tuning last layers had lower MAE
Multi-task Learning	[39]	User-as-task and cluster-as-task based on gender and personality cluster • Traditional model	• Self/Closed • In-the-wild • P, B • Mood, health, stress	Investigate the impact of the number of clusters both in user-dependent (for each participant, 60% as train, 20% as validation, and 20% as test set) and user-independent (each participant to one of the train, validation, test set) setting	For both settings when the number of clusters is equal to the number of users in the training set, i.e., user-as-task, resulted in lowest MAE
	[65]	User-as-task • DL model	• Open data [23] [73], Self-collected • Controlled, In-the-wild • P • Stress	Compare personalized and single-task model (for each participant, 80% as train and 20% as test)	For [23], personalization increased AUROC and kappa For [73], personalization increased AUROC and kappa For self-collected, personalization increased AUROC and kappa
	[86]	User-as-task and cluster-as-task based on gender and personality cluster • Traditional model, DL model	• Self/Closed [66] • In-the-wild • P, B, C • Mood, health, stress	Compare personalized and single-task model (for each participant, 60% as train, 20% as validation, 20% as test set)	Personalization decreased MAE and increased f1 score, especially with user-as-task rather than cluster-as-task
	[61]	User recognition and stress detection as tasks • DL model	• Open data [67] • Controlled • P, B • Stress	Compare personalized and generalized model using LOPO CV	Personalization increased f1 score
	[74]	Moods-as-task and cluster-as-task based on gender and personality cluster • Traditional model, DL model	• Self/Closed [66] • In-the-wild • P, B, C • Mood, health, stress	Compare personalized and single-task model (for each participant, 8% as train, 20% as test set)	Personalization increased accuracy and AUROC, especially with cluster-as-task rather than moods-as-task
Others	[70]	Modify SVM objective function by adding participant-specific parameter	• Self/Closed • Controlled • P • Stress	Compare personalized and generalized model using LOPO CV	Suggested algorithm increased precision value at 80% recall

research, models initially pre-trained on extensive group data are employed to capture broadly useful representations. These models are then fine-tuned to identify features uniquely characteristic of the target user [41]. Yu et al. [87] pre-trained a deep LSTM model for wellbeing prediction on all other users' data and fine-tuned the last LSTM and the final layers using the target user's data, showing a lower MAE than the generalized model even with only 10% of the target user's data. Kathan et al. [29] compared the MAE of fine-tuned and generalized models, finding little difference in depression prediction for the same day but a reduction of about 0.2 in MAE for one day ahead forecasting.

MTL aims to improve the performance of each task by learning multiple related tasks simultaneously and sharing representations [10]. In this approach, representation sharing between tasks can be realized by sharing layers between tasks in deep learning networks or by applying similarity constraints to the weights of classifiers [10, 27]. Personalization has been achieved by defining users or clusters as tasks and creating models specific to each individual or cluster [74]. Saeed et al. [65] created a driver's stress detection model using HR and EDA data, defining individual users as tasks, and compared the performance of the multi-task neural network (NN) model with a single-task NN model. They found that multi-task models showed an accuracy improvement of about 1–6% over the single-task model across three different datasets. Taylor et al. [74] created a wellbeing prediction MTL model by clustering users based on personality and gender and defining each cluster as a task. They found that defining clusters as tasks allowed for more accurate predictions for new participants not seen during the training process, with an accuracy improvement of about 11-21% compared to a single-task model.

Beyond data- and model-level classifications, Ferrari et al. [16] presented a spectrum of personalization based on how a target unseen user's data is used for personalization; i.e., *user-dependent personalization* (= user-specific personalization) where only a target unseen user's data is used for training and testing, and *hybrid personalization* where part of a target unseen user's data along with all the other users' data is used for model training. Further, when a target unseen user's data is not used for model training, that approach is called *user-independent modeling*. In this work, we extend this concept to classify personalization techniques based on whether a target unseen user's data is used for personalization: unseen user-dependent vs. user-independent models. *Unseen user-dependent models* include user-specific personalization, fine-tuning of user-independent models using an unseen user's data, and hybrid personalization. In contrast, *unseen user-independent models* do not use an unseen user's data for personalization by assuming that there are *similar people* like the unseen user (i.e., *user groups* to which the unseen user belongs in the training data). We call this approach *group-based personalization*. This concept is similar to "collaborative filtering," where responses by similar users are used for item recommendation (e.g., same personality traits or gender). For unseen user-independent modeling, we can build *separate models* for each user group (or a cluster of similar users) or a unified multi-task model (e.g., cluster as a task).

Note that it is crucial to consider the practical challenges of building user-dependent models. Such models often rely on a *user-in-the-loop personalization* approach, where systems prompt users to label their current affective states [13, 48]. While this can lead to highly personalized outcomes, the burden of frequent labeling in everyday contexts can become overwhelming for users [78]. To address this, recent advances in adapting models for new participants have shown promising results. By treating users as tasks, meta-learning (or few-shot learning) enables models to rapidly adapt to a new user with only a few labeled data points. Meta-learning implements transfer learning with many source tasks to solve new tasks using only a few labeled data points from the new tasks (known as few-shot adaptation). This approach's effectiveness has been demonstrated in sensor-based human activity recognition [20], video-based physiological measurement [42], and emotion/depression recognition [83, 90]. In general, we can use domain adaptation where target domain (i.e., person) data are used for training (which belongs to transductive learning) [11]. In particular, *unsupervised* domain adaptation uses only *unlabeled* target domain data; this means that a target user's passive sensor data is used for training, but no user involvement is required for labeling. Typical domain adaptation methods handle differences in input data distributions via data reweighting, feature alignment, or domain translation [35]. Recent studies highlighted the potential benefits of unsupervised domain adaptation for activity and mood classification [11, 47]. Additionally, recent domain generalization techniques showed that such domain shifts can be simulated during training [40], thereby obviating the need to use target data at all. While recent studies explored advanced ML approaches, our focus remains on well-known techniques to systematically evaluate whether they demonstrate consistent results, addressing the personalization challenges by establishing baselines.

2.3 Reproducible Research and Open Datasets for Affect Recognition

Given that the goal of machine learning research is to develop algorithms capable of reliably solving large-scale complex problems, the importance of reproducibility is increasingly emphasized [3, 21, 45]. Recently, McDermott et al. [45] stated that for ML research to be fully reproducible, it must meet three reproducibility criteria. First is *technical reproducibility*, which requires the ability to fully replicate the exact results reported in a paper technically. This necessitates the release of the code and dataset used in the paper, including sufficient details to run them correctly. Second is *statistical reproducibility* that minor numerical differences in results due to processes like resampling should not statistically significantly impact the main results or conclusions of the research. This represents *internal validity*, which requires detailed reporting of randomized trials, such as listing both the mean and the standard deviation of performance metrics over several random initializations. Third is *conceptual reproducibility*, which entails obtaining results under new conditions that align with the original experiment's theoretical explanations, representing *external validity*. For example, model evaluation with multiple datasets can demonstrate that the presented model can properly adapt to new, previously unseen data.

As AI-powered healthcare tools, which can directly impact human health, become more prevalent, reproducible results in machine learning for health research are in the public interest [45]. However, reproducibility was not thoroughly considered in prior personalized affect recognition research. More than half of the studies listed in Table 1 and 2 evaluated proposed methods using self-collected, unpublished (closed) datasets. Only one paper conducted evaluations using more than two datasets [65]. Additionally, a lack of detailed descriptions of actual implementation, such as hyperparameter settings, is prevalent, and only two studies have open-sourced their code [2, 74]. Consequently, this might complicate the process of reproducing the research results.

This work aims to evaluate well-known personalization techniques in affect recognition. We evaluate four different personalization methods using five open datasets to verify their *conceptual reproducibility* as in recent work on reproducible stress detection [89]. We considered open datasets, which include physiological and behavioral signals collected in a controlled environment, as listed in Table 3. We also open-source the entire process code, from data preprocessing to final evaluation, to ensure that it is *technically reproducible*, and other researchers can use our code to evaluate their personalization techniques rigorously.

3 Methods

3.1 Datasets

In our evaluation, we used five open datasets in Table 3: AMIGOS, ASCERTAIN, WESAD, CASE, and K-EmoCon. These datasets were chosen because they include the necessary user profile survey information necessary for identifying a group of individuals similar to the unseen user, which is essential for evaluating cluster-specific and multi-task learning personalized models. The other datasets were excluded due to the lack of this profile survey information. Regarding emotion annotations, AMIGOS, ASCERTAIN, CASE, and K-EmoCon utilize a self-report approach for emotion annotation, while WESAD employs a stimulus-based approach. Specifically, AMIGOS and ASCERTAIN provide a single self-report value for each stimulus (i.e., video), whereas CASE and K-EmoCon offer continuous annotation throughout the entire experiment. Detailed information about each dataset and data source can be found in Appendix A and Table 12, respectively.

3.2 Preprocessing

For the purpose of using each sensor signal in an end-to-end DL system, a preprocessing procedure was carried out. As proposed by Dzieżyc et al. [14], a sequence of winsorization, filtering, downsampling, normalization, and segmentation was applied. The resulting data were used as the input for the model. By adopting a participant-dependent approach in all preprocessing steps, where each participant's data was processed individually, we

Table 3. Open datasets including physiological or behavioral signals collected in a controlled environment.

Dataset	Signal	Label	# of Ps	Duration	Profile Survey
DEAP [33] (2012)	• Physiological (EEG, EDA, BVP, RESP, TEMP, EMG, EOG)	• Per stimuli annotation • SAM	32	40 mins; 40 videos	Not specified
SWELL [34] (2014)	• Physiological (ECG, EDA)	• Per stimuli annotation • SAM, stress	25	3 hours	Not included
DECAF [1] (2015)	• Physiological (MEG, hEOG, ECG, tEMG)	• Per stimuli annotation • PANAS	30	88 mins; 75 videos	Not included
ASCERTAIN [72] (2016)	• Physiological (ECG, EDA, EEG) • Behavioral (ACC)	• Per stimuli annotation • Arousal, valence, engagement, liking, familiarity	58	50 mins; 36 videos	Big-Five personality traits
DREAMER [30] (2017)	• Physiological (EEG, ECG)	• Per stimuli annotation • Arousal, valence, dominance	23	1 hour; 18 videos	Not included
WESAD [67] (2018)	• Physiological (RESP, ECG, EDA, EMG, TEMP) • Behavioral (ACC)	• Per stimuli annotation • PANAS, STAI, SAM, SSSQ	15	1 hour	Age, gender
CASE [69] (2019)	• Physiological (ECG, RESP, BVP, EDA, TEMP, EMG)	• Continuous annotation • Arousal, valence *# of labels per participant: 49,000*	30	21 mins; 8 videos	Age, gender
K-EmoCon [58] (2020)	• Physiological (EEG, ECG, BVP, EDA, TEMP) • Behavioral (ACC)	• Continuous annotation • Arousal, valence, affective categories *# of labels per participant: 120–180*	32	10 mins	Age, gender
AMIGOS [50] (2021)	• Physiological (EEG, ECG, EDA)	• Per stimuli annotation • PANAS, SAM, Ekman's basic emotions	40	79 mins; 20 videos	Big-Five personality traits

aligned with the findings of Kathan et al. [29] and Tervonen et al. [77], who demonstrated that such personalized processing significantly enhances performance outcomes. Figure 2 shows the overall steps for data preprocessing.

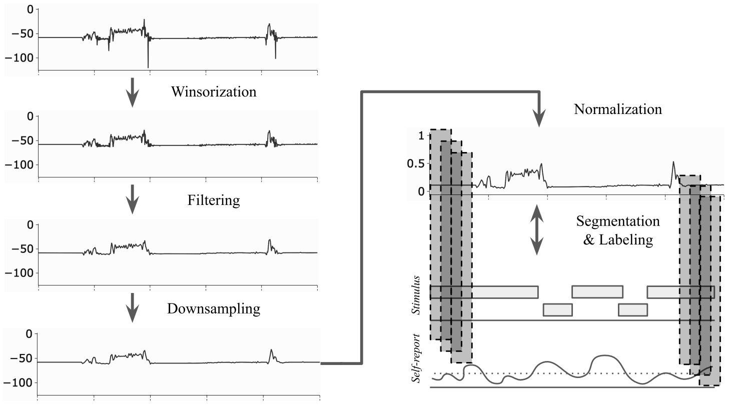


Fig. 2. Overview of Data Preprocessing.

Firstly, outliers in the upper and lower 3% range were removed for each sensor signal. Subsequently, a Butterworth low-pass filter with a 10 Hz cut-off was applied to eliminate signals above 10 Hz. Next, to prevent an excessive number of parameters in deep learning models, downsampling was performed on each sensor

Table 4. Summary of Data Used for Further Analysis.

Dataset	**Used Signal**	**Label**	**Label Distribution**	**Used # of Ps (N)**	**Used Duration**	**Average # of Segments per Ps**	**Used Survey**
AMIGOS	*Total: 20* EEG (14) ECG (2) EDA (1) ACC (3)	Arousal Valence (Self-report based)	50:50 48:52	31	22.6mins	292.00	Big-five personality, Age, Gender
ASCERTAIN	*Total: 6* ECG (2) EDA (1) ACC (3)	Arousal Valence (Self-report based)	50:50 47:53	58	30mins	357.55	Big-five personality
WESAD	*Total: 14* Chest ECG Chest ACC (3) Chest EMG Chest EDA Chest TEMP Chest Resp Wrist BVP Wrist ACC (3) Wrist EDA Wrist TEMP	Stress (Stimulus based)	64:36	15	16.5mins	207.20	Age, Gender
CASE	*Total: 8* ECG BVP EDA RESP TEMP EMG (3)	Arousal Valence (Continuous Self-report based)	57:43	30	21mins	263.10	Age, Gender
K-EmoCon	*Total: 6* EDA ACC (3) TEMP BVP	Arousal Valence (Continuous Self-report based)	64:36	21	10mins	125.62	Age, Gender

signal, referencing the rate from previous literature [14]. The original sampling rate and the post-downsampling rate for each dataset are presented in Appendix B. Following this, normalization was applied. Normalization typically involves adjusting data either by scaling it according to the maximum and minimum values of that specific participant's data (i.e., min-max normalization) or by subtracting the mean and then dividing by the standard deviation [8]. Min-max normalization was used to bring all the signals into the same range of 0–1, as in the previous works [14, 39]. The final step involved segmentation, where all sensor signals were divided in a fixed time length. When using autonomic nervous system activity measures such as BVP, EDA, and TEMP, a common window size ranges from 10 to 30 or 60 seconds [36]. Prior work showed mixed findings on the effect of window size in that best-performing window sizes varied based on ML models and evaluation methods. When considering a scenario of real-time affect state detection using wearable sensors, researchers typically use a 10-second window [12, 18, 54, 55]. In line with this practice, our study adopted a 10-second window with 50% overlapping. As a result, on average, AMIGOS generated 292.0 windows, ASCERTAIN 357.6 windows, WESAD 207.2 windows, CASE 263.20 windows, and K-EmoCon 125.62 windows per participant.

Subsequently, affect labels were assigned to each window. Stimulus-based labeling was employed for WESAD, which underwent labeling only once after each experimental session (non-stressed for amusement vs. stressed for social/mental stressors) (see the scenario in Figure 12). Meanwhile, AMIGOS, ASCERTAIN, CASE, and K-EmoCon had self-report labels. For fair comparisons across different datasets, we considered a labeling process by setting a threshold for the binarization of annotation responses. Although many studies used an absolute value threshold for binarization, recent studies have employed personalized, participant-specific thresholds to address the individual differences in subjective self-report responses [12, 32]. The binarization threshold for each participant was calculated using the average of all self-reported values provided by each participant. Then, if the average label values within each window were less than the computed threshold, a label of low affect was assigned; otherwise, a label of high affect was assigned. Specifically for AMIGOS and ASCERTAIN, each video had a rating, resulting in all segments of each video being labeled with the corresponding label.

Table 4 summarizes the data used for further analysis. All the physiological and behavioral signals, except for the EEG signal in ASCERTAIN and K-EmoCon dataset, are utilized for practical considerations. EEG in ASCERTAIN dataset was excluded due to inconsistent collection frequency across users, and in the K-EmoCon dataset, it was excluded due to its significantly lower sampling rate than reported. Participants with issues in data completeness and unnecessary experimental periods such as baseline, meditation, and rest were excluded, and the table indicates the final number of participants and duration used.

3.3 Non-Personalized Model

We created a non-personalized model as a baseline to investigate whether existing personalized methods demonstrate a significant performance benefit. To build such models, we employed **three different DL architectures**, namely the Fully Convolutional Network (FCN), Residual Network (ResNet), and Multi-Layer Perceptron with LSTM (MLP-LSTM) throughout all methods. Dzieżyc et al. [14] compared ten end-to-end architectures for emotion classification using four physiological signal datasets and found that FCN and ResNet demonstrated the best performance. The inclusion of LSTM was justified as it remains one of the most commonly used architectures in DL-based emotion recognition systems research [43]. In the given architectures, individual signal channels are treated as separate branches, each consisting of stacked layers. These branches eventually come together to generate the final output prediction of the model, as shown in Figure 3. Detailed explanations for each architecture can be found in prior studies [14, 25].

Leave-one-participant-out evaluation: Each DL architecture was trained using all participants' data except for the target participant's and then evaluated on the target participant's data. Out of a total of $\mathcal{N}$ participants in each dataset, 1 participant was designated as the target. The remaining $\mathcal{N}$-1 participants were used for the model training. This process was iteratively repeated for all $\mathcal{N}$ participants, designating each as the test participant in turn. Finally, the average and the standard deviation of all results were computed.

In Figure 4, we illustrate the training and evaluation process of a non-personalized model. For the sake of illustration, in the figure, we assumed that there are five participants. We denoted the data from the target participant as $\mathrm{D}_{\mathcal{T}}$, and the remaining four participants as D_1 to D_4.

3.4 Personalized Models

Our focus is on developing personalized models for unseen users who have limited available data, i.e., sensor signals and corresponding labels. This addresses the practical challenge of acquiring a substantial amount of data for new users. Consequently, we exclude user-specific model that necessitates a substantial amount of labeled data for each individual. Approaches to building personalized models for unseen users can be classified based on whether they involve using limited data of these users during the training process. In other words, if a model uses some part of an unseen user's data, it is classified as an (unseen) *user-dependent* approach; otherwise, it is an (unseen) *user-independent* approach.

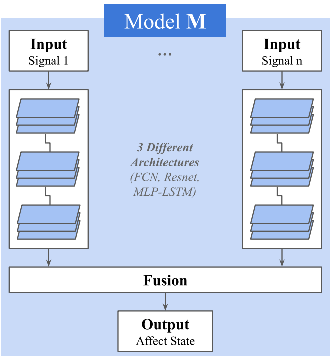


Fig. 3. DL architectures

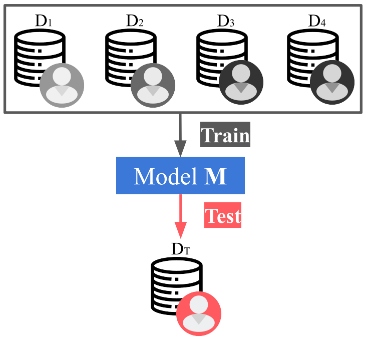


Fig. 4. Training and Evaluation Process of Non-Personalized Model

The fine-tuning model represents a user-dependent approach, where a certain amount of unseen users' data is utilized to refine a non-personalized model. Alternatively, a hybrid model that is trained using the data of an unseen user along with the data from $\mathcal{N}$-1 other users also falls under the user-dependent category. Cluster-specific and multi-task learning models can be used to enable a user-independent approach. These methods can train models without using the unseen user's data. As shown later, we can leverage participant profile information to identify similar users, and the unseen users can use the existing models trained for similar users based on user profiles. Note that it is also feasible for cluster-specific and multi-task learning models to implement a *user-dependent approach* by integrating the unseen user's data into the training (as if it belongs to a similar user). In this work, we only consider a user-independent approach for constructing and evaluating these methods. This enables us to examine whether personalization methods are applicable to unseen users without any use of personal data except profile information, which contrasts with user-dependent approaches, such as fine-tuning and hybrid models. The following subsections provide detailed explanations of each method.

3.4.1 Unseen User-Dependent. We consider fine-tuning and hybrid modeling for unseen user-dependent models. Figure 5 and 6 show the training and evaluation process of a fine-tuning and hybrid model, respectively.

Fine-Tuning: The training process proceeds as follows. Initially, all layers of the network undergo pre-training with data from $\mathcal{N}$-1 participants, establishing a foundational understanding of the task. This is followed by retraining (i.e., tuning) the network using a small number of data from the target participant, aiming to create a model that is tailored specifically to that individual while avoiding overfitting. To ensure a balanced dataset for tuning, we used a specific number of data points from each label in the target participant's data. Also, given the high temporal dependency inherent in time series data, selecting data points randomly from the entire range could result in a biased evaluation. Therefore, we chose the initial sequence of data points from each label. Here, the tuning process can be applied either *to the whole network* or *limited to the final layer*. It is known that the final layer of deep neural networks greatly depends on the chosen dataset and task [85]. Thus, by retraining just the final output layer, the model can be more effectively tailored to reflect the unique attributes of the target participant. Moreover, in line with approaches in prior studies [7, 87], we examined the impact of *varying the number of data used* for this fine-tuning phase. We evaluated the model's performance using different quantities, 20%, 30%, 40%, and 50% of a target participant's data. The final step involves testing the fine-tuned model with the remaining data points of the target participant. For a given dataset, this procedure was repeated for each participant, treating each one in turn as the target.

Hybrid (Partially Personalized): Recent studies have proposed a hybrid model known as the partially personalized approach for personalization, which is similar to, yet distinct from, traditional fine-tuning [46, 76]. Contrary to fine-tuning, where a pre-trained model is adjusted with the target participant's data, this approach simultaneously utilizes data from N-1 participants and the target participant to train the model. As a result, the model undergoes training only once, other than the twice-required training in fine-tuning. In line with these prior works, the training process unfolds as follows. The network's layers are trained using the data from N-1 participants and 50% of the data from the target participant [46, 76]. The remaining 50% of the target participant's data is then used to test the trained model. Similar to the previous method, this process is iteratively conducted for each participant as the target.

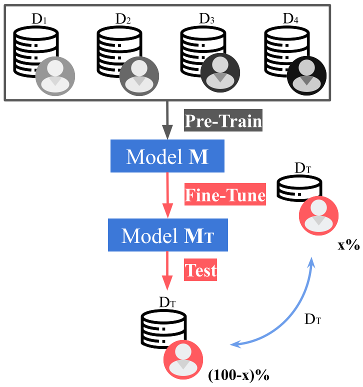


Fig. 5. Training and Evaluation Process of Fine-Tuning Model

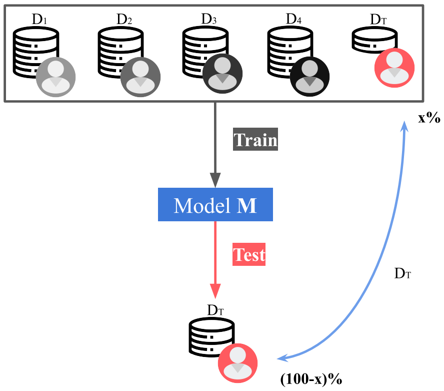


Fig. 6. Training and Evaluation Process of Hybrid Model

3.4.2 *Unseen User-Independent.*

We consider cluster-specific and multi-task learning models for user-independent personalization. Figure 7 and 8 show the training and evaluation process of a cluster-specific and multi-task learning personalized model, respectively.

Cluster-Specific: The cluster-specific method is rooted in the assumption that building a model using only data from 'similar' participants to the target unseen participant would be more helpful than using data from all the other participants. It is important to note that there is no architectural difference from the generalized model, but the difference lies in the data used for training.

In order to find similar participants, previous research employed demographics or psychological information such as gender, age, and personality traits and performed K-Means clustering on participants [29, 39, 74]. The number of clusters, K, used in prior studies varied—it was sometimes a fixed value or determined by the highest mean Silhouette score [9, 29, 77]. In our study, we similarly applied K-Means clustering, utilizing age and gender information for WESAD, K-EmoCon, and CASE. For AMIGOS and ASCERTAIN, we utilized personality information derived from the Big Five Inventory questionnaire [59].

The training is initiated by creating clusters based on trait information of N-1 participants, leading to the development of a distinct model for each cluster. Only the participants within the same cluster were used for training their respective models. For testing, the target participant's cluster was identified using his or her trait information (i.e., age/gender or personality), and the model corresponding to that cluster was used for evaluation. This approach ensured that the target participant's physiological and label data remained *unseen* until the final testing phase. The procedure was repeated for each subject in the datasets, treating them as the target in turn.

Moreover, we explored the *impact of varying the number of clusters*. We compared performances using fixed K values ranging from 2 to 5 and dynamically calculated K values based on the Silhouette score. We limited K values to 5 because larger values resulted in clusters with too few participants due to the small overall number of participants in each dataset.

Multi-task Learning: Multi-task Learning (MTL) is a learning method that simultaneously trains on multiple similar tasks by sharing information between them. In the context of DL, MTL is implemented through a combination of shared layers which are common across all tasks, and task-specific layers which are unique to each task. This structure not only allows for the acquisition of general knowledge applicable across various tasks but also supports tailored learning for individual tasks [10].

As in the previous research, we compared the *distinct task definitions*: i.e., user-as-task and cluster-as-task [39, 65, 74, 86]. Both approaches train all network layers except the last fully connected (FC) layer and the output layer, using data from $\mathcal{N}$-1 participants. In the user-as-task approach, the last FC layer and output layer are trained using each participant's data (excluding the target), calculating weights for each participant. Then, we find the participant who is the most similar to the target and apply weights trained on that participant. Likewise, in the cluster-as-task approach, the last FC layer and output layer are trained using all the participants' data in each cluster, calculating weights for each cluster. We then find the cluster the target participant belongs to and apply the weights trained on that cluster to the target. In both approaches, the similarity was determined using the demographics or psychological information, just as in the cluster-specific method. The number of clusters K in the cluster-as-task approach was determined based on the highest mean Silhouette score, a method most commonly preferred in previous research [39, 74, 87]. Finally, using the unseen target participant data, we tested the model, repeating this for all $\mathcal{N}$ participants.

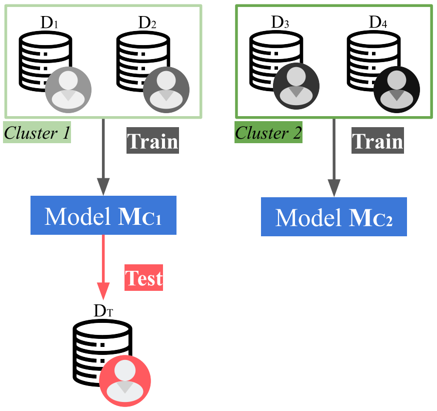


Fig. 7. Training and Evaluation Process of Cluster-Specific Model

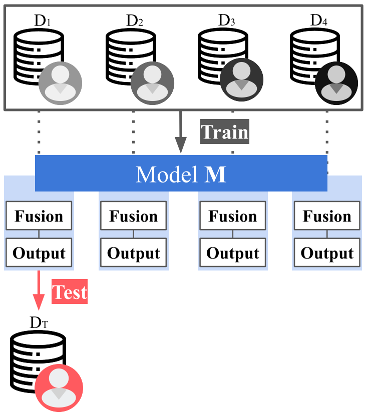


Fig. 8. Training and Evaluation Process of Multi-task Learning Model (User-as-task)

3.5 Evaluation Methods

In each dataset, every participant was designated once as the target, forming the test set. The others, excluding the target, were randomly divided into training and validation sets at an 80:20 ratio [29]. To ensure a fair comparison among the methods, the DL architecture-related hyperparameters were fixed in line with those used in previous research on DL for time series classification [25]. These settings are detailed in the accompanying code. Notably, the validation set's role was not to tune the architecture-related hyperparameters but to prevent overfitting on the training set. A maximum of 100 epochs was set for the training, with early stopping implemented if there was no improvement in the validation loss for 15 consecutive epochs. This procedure was replicated for each participant, treating each in turn as the target. As mentioned earlier, the average and standard deviation of the results were calculated and reported, providing a detailed assessment of the model's performance.

As metrics, we report accuracy, f1-score, and AUROC. While accuracy is the most commonly used metric, it is a poor indicator for imbalanced datasets and does not account for the trade-off between precision and recall; hence, we also report f1-score, the harmonic mean of precision and recall [24]. Following previous literature, we use macro f1-score, which ensures equal importance is given to each class by computing the average of their individual performances when each is treated as the positive class [84]. Meanwhile, AUROC is known to be more informative than accuracy and has been used in previous studies when comparing learning algorithms [24]. Therefore, we report all these metrics but mainly use AUROC to compare the performance of different personalization methods.

3.6 Implementation

All models were optimized using the Adam algorithm, with a learning rate of 0.003 and a weight decay of 1e-6. For optimal training, we employed TensorFlow's ReduceLROnPlateau callback function, which adjusts the learning rate when the validation loss ceases to decrease, thereby inducing model improvement. Implementation of non-personalized model code is based on code provided at https://github.com/Emognition/dl-4-tsc.git. Building upon this foundation, we have developed and open-sourced comprehensive preprocessing and personalized model code using the TensorFlow framework, which is available at a Github repository.[2] Please refer to the README.md file for detailed information.

4 Results

We start by presenting the performance of non-personalized models across datasets. Then, we detail the outcomes for each of the four personalization techniques. For each technique, we report the results of the various experiments conducted, focusing on the key findings and performance metrics. In the latter part, we compare the best results from each personalization technique against the non-personalized counterpart. This offers a comprehensive overview of how each personalization approach works in relation to the non-personalized one.

4.1 Non-Personalized Model

Table 5 reports the performance of non-personalized models evaluated on four distinct datasets. The WESAD dataset, which utilized stimulus-based labeling, demonstrated a significantly higher detection performance. In contrast, the remaining four datasets, which employed self-report based labeling, exhibited relatively lower performance. This observation is consistent with prior studies, which found that using self-reported data for stress or emotion detection tends to yield reduced performance [52, 79].

4.2 Analysis of Personalization Techniques

4.2.1 Fine-Tuning: Amount of Layers Tuned and Data Used. Figure 9 plots the changes in AUROC for fine-tuned models under different settings, along with non-personalized models, namely *without tuning*. The result revealed no consistent patterns regarding the number of layers tuned, as the most effective strategy for attaining the highest performance was dependent on the specific dataset and model architecture. Furthermore, increasing the volume of data used for tuning did not consistently lead to enhanced performance. However, in comparison to their non-personalized counterparts, most of the dataset-architecture combinations exhibited at least one scenario where a fine-tuned model achieved higher AUROC values, underscoring the beneficial impact of fine-tuning. The

[2] https://github.com/Kaist-ICLab/Personalized_Affective_Computing.git

Table 5. Non-Personalized Model Evaluation Results (Average with Standard Deviation).

Dataset	Architecture	Accuracy	F1-score	AUROC	Dataset	Architecture	Accuracy	F1-score	AUROC
AMIGOS (Arousal)	FCN	0.488 (0.105)	0.403 (0.090)	0.500 (0.100)	**AMIGOS** (Valence)	FCN	0.470 (0.111)	0.373 (0.104)	0.518 (0.131)
	MLP-LSTM	0.481 (0.117)	0.332 (0.058)	0.504 (0.100)		MLP-LSTM	0.478 (0.122)	0.327 (0.060)	0.476 (0.109)
	ResNet	0.474 (0.111)	0.375 (0.100)	0.490 (0.115)		ResNet	0.496 (0.100)	0.398 (0.102)	0.493 (0.106)
ASCERTAIN (Arousal)	FCN	0.503 (0.062)	0.372 (0.064)	0.511 (0.071)	**ASCERTAIN** (Valence)	FCN	0.542 (0.064)	0.379 (0.063)	0.514 (0.060)
	MLP-LSTM	0.506 (0.062)	0.342 (0.045)	0.498 (0.035)		MLP-LSTM	0.540 (0.071)	0.349 (0.031)	0.496 (0.047)
	ResNet	0.511 (0.061)	0.400 (0.080)	0.506 (0.070)		ResNet	0.532 (0.071)	0.397 (0.082)	0.520 (0.064)
WESAD (Stress)	FCN	0.839 (0.187)	0.786 (0.256)	0.915 (0.203)	-	-	-	-	-
	MLP-LSTM	0.898 (0.177)	0.874 (0.223)	0.922 (0.195)		-	-	-	-
	ResNet	0.805 (0.231)	0.769 (0.272)	0.906 (0.196)		-	-	-	-
CASE (Arousal)	FCN	0.550 (0.106)	0.461 (0.124)	0.646 (0.165)	**CASE** (Valence)	FCN	0.512 (0.147)	0.385 (0.153)	0.651 (0.159)
	MLP-LSTM	0.519 (0.092)	0.339 (0.040)	0.508 (0.069)		MLP-LSTM	0.561 (0.106)	0.356 (0.045)	0.548 (0.134)
	ResNet	0.557 (0.116)	0.469 (0.146)	0.648 (0.155)		ResNet	0.536 (0.138)	0.445 (0.156)	0.620 (0.169)
K-EmoCon (Arousal)	FCN	0.526 (0.142)	0.475 (0.139)	0.505 (0.176)	**K-EmoCon** (Valence)	FCN	0.558 (0.106)	0.484 (0.092)	0.507 (0.147)
	MLP-LSTM	0.514 (0.135)	0.422 (0.118)	0.523 (0.173)		MLP-LSTM	0.494 (0.156)	0.391 (0.126)	0.520 (0.174)
	ResNet	0.514 (0.131)	0.450 (0.126)	0.487 (0.188)		ResNet	0.487 (0.110)	0.422 (0.096)	0.508 (0.130)

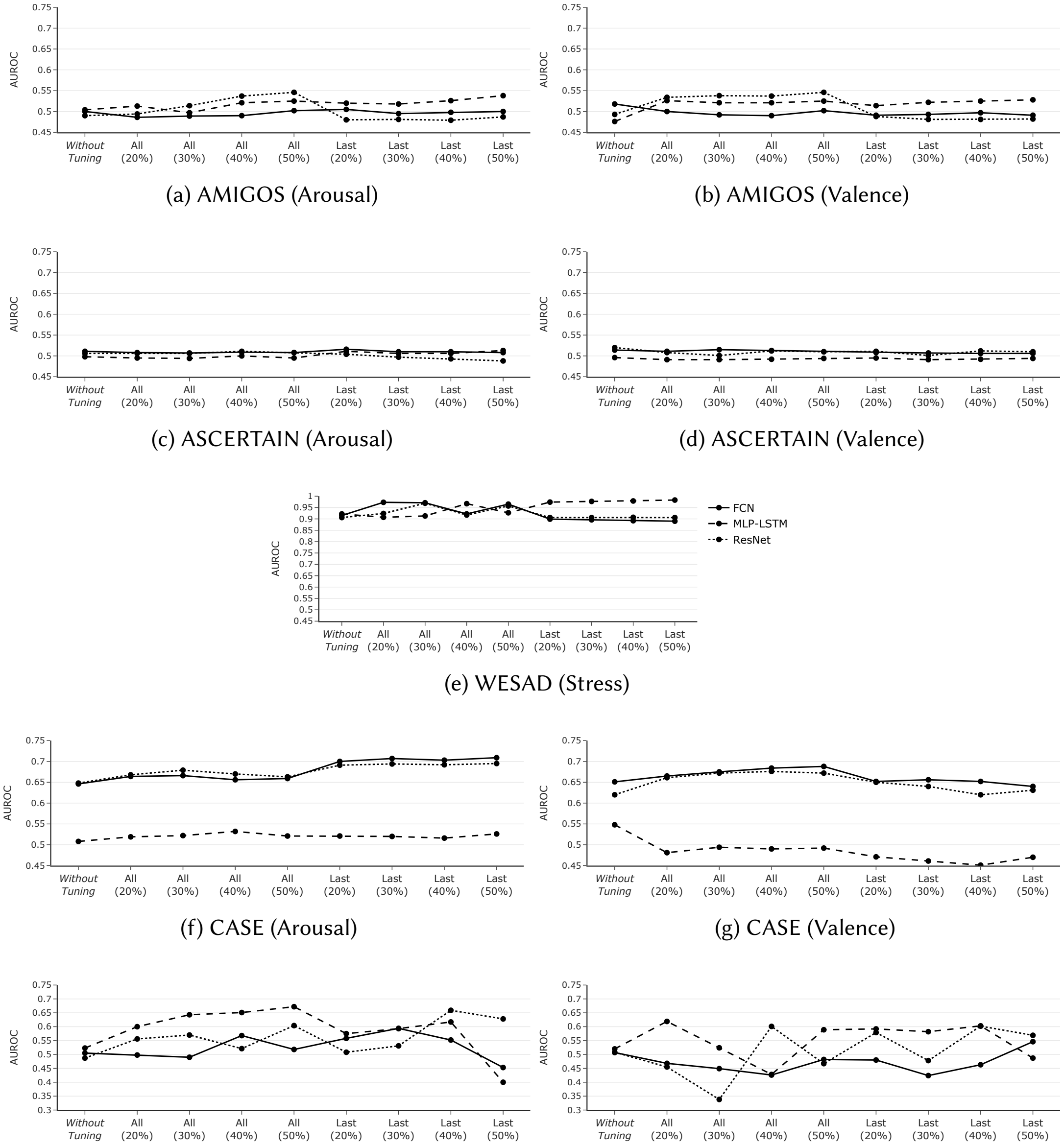
detailed numerical results corresponding to the AUROC values depicted in Figure 9 and other metrics, accuracy, and f1-score, are comprehensively documented in Appendix C.

4.2.2 Hybrid. Table 6 reports the performance of hybrid models using 50% of the test user's data evaluated on four datasets. Each dataset-architecture pair showed varying performances compared to the results of the non-personalized model in Table 7, with both higher and lower outcomes.

Table 6. Results for Hybrid Models.

Dataset	Architecture	Accuracy	F1-score	AUROC	Dataset	Architecture	Accuracy	F1-score	AUROC
AMIGOS (Arousal)	FCN	0.473 (0.163)	0.383 (0.133)	0.512 (0.151)	**AMIGOS** (Valence)	FCN	0.474 (0.170)	0.368 (0.128)	0.494 (0.145)
	MLP-LSTM	0.509 (0.178)	0.358 (0.102)	0.476 (0.107)		MLP-LSTM	0.475 (0.186)	0.322 (0.087)	0.511 (0.142)
	ResNet	0.537 (0.161)	0.410 (0.118)	0.518 (0.136)		ResNet	0.511 (0.173)	0.392 (0.122)	0.515 (0.112)
ASCERTAIN (Arousal)	FCN	0.508 (0.085)	0.393 (0.079)	0.508 (0.067)	**ASCERTAIN** (Valence)	FCN	0.554 (0.084)	0.385 (0.076)	0.505 (0.075)
	MLP-LSTM	0.505 (0.088)	0.342 (0.055)	0.491 (0.056)		MLP-LSTM	0.547 (0.084)	0.352 (0.036)	0.499 (0.035)
	ResNet	0.498 (0.082)	0.391 (0.082)	0.505 (0.085)		ResNet	0.543 (0.083)	0.381 (0.064)	0.515 (0.066)
WESAD (Stress)	FCN	0.923 (0.161)	0.909 (0.168)	0.976 (0.074)	-				
	MLP-LSTM	0.885 (0.233)	0.863 (0.261)	0.913 (0.212)					
	ResNet	0.868 (0.215)	0.835 (0.234)	0.979 (0.066)					
CASE (Arousal)	FCN	0.594 (0.139)	0.490 (0.146)	0.655 (0.197)	**CASE** (Valence)	FCN	0.562 (0.180)	0.494 (0.182)	0.655 (0.217)
	MLP-LSTM	0.543 (0.152)	0.346 (0.066)	0.520 (0.106)		MLP-LSTM	0.563 (0.194)	0.350 (0.089)	0.506 (0.089)
	ResNet	0.617 (0.142)	0.511 (0.147)	0.646 (0.168)		ResNet	0.548 (0.170)	0.440 (0.160)	0.651 (0.200)
K-EmoCon (Arousal)	FCN	0.520 (0.355)	0.372 (0.290)	0.509 (0.358)	**K-EmoCon** (Valence)	FCN	0.429 (0.329)	0.269 (0.174)	0.443 (0.202)
	MLP-LSTM	0.453 (0.384)	0.362 (0.353)	0.650 (0.373)		MLP-LSTM	0.407 (0.447)	0.341 (0.410)	0.752 (0.255)
	ResNet	0.455 (0.308)	0.385 (0.308)	0.594 (0.312)		ResNet	0.465 (0.322)	0.296 (0.164)	0.643 (0.349)

4.2.3 Cluster-Specific. We evaluate the cluster-specific models by varying the number of clusters. Figure 10 illustrates the changes in AUROC for cluster-specific models with different K settings, along with non-personalized models, i.e., *without clustering*. In most cases, fixing the value of K exhibited better performance than dynamically calculating it using the highest mean silhouette score. The optimal fixed value of K varied depending on the dataset and the model used. Moreover, it is observed that except for the AMIGOS and ASCERTAIN dataset, cluster-specific models demonstrate lower AUROC values compared to one-size-fits-all non-personalized ones. In AMIGOS and ASCERTAIN, most of combinations of dataset and architecture showed instances where a cluster-specific model attained superior AUROC values, highlighting its positive effect. It should be noted that, in contrast to other datasets which were clustered according to age and gender, AMIGOS and ASCERTAIN involved



(a) AMIGOS (Arousal)

(b) AMIGOS (Valence)

(c) ASCERTAIN (Arousal)

(d) ASCERTAIN (Valence)

(e) WESAD (Stress)

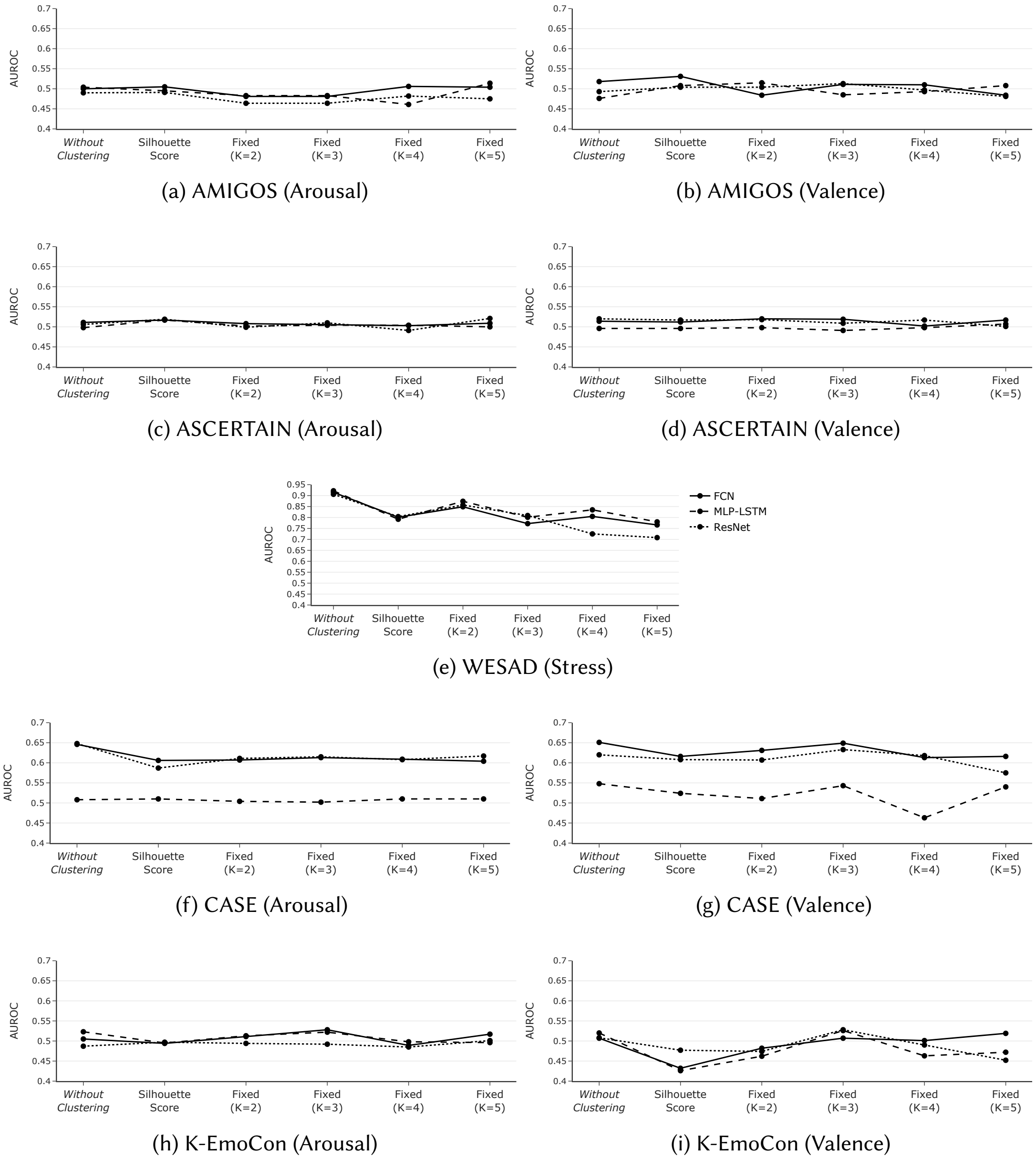
(f) CASE (Arousal)

(g) CASE (Valence)

(h) K-EmoCon (Arousal)

(i) K-EmoCon (Valence)

Fig. 9. Fine-Tuning: AUROC Across Various Settings. Line styles represent different deep learning architectures: a solid line for FCN, a dashed line for MLP-LSTM, and a dotted line for ResNet. For the vertical scale, we used a default range of 0.45 to 0.75, except for the WESAD dataset, where we employed a range of 0.45 to 1.00 and for the CASE data, a range of 0.30 to 0.75.



(a) AMIGOS (Arousal)

(b) AMIGOS (Valence)

(c) ASCERTAIN (Arousal)

(d) ASCERTAIN (Valence)

(e) WESAD (Stress)

(f) CASE (Arousal)

(g) CASE (Valence)

(h) K-EmoCon (Arousal)

(i) K-EmoCon (Valence)

Fig. 10. Cluster-Specific: AUROC Across Various Settings. For the vertical scale, we used a default range of 0.4 to 0.7, except for the WESAD dataset, where we employed a range of 0.4 to 1.0.

clustering based on personality. Appendix C thoroughly presents the detailed numerical outcomes, including AUROC values, accuracy, and f1-scores, corresponding to the data shown in Figure 10.

4.2.4 Multi-task Learning. Figure 11 shows the changes in AUROC for multi-task learning models with two different settings (i.e., user- and cluster-as-task), along with non-personalized models (i.e., *one-size-fits-all*). No distinct trends were observed, as the optimal task definition for achieving the highest performance varied depending on the dataset used and the specific model architecture employed. Generally, it is noted that multi-task learned models tend to show lower AUROC values compared to non-personalized ones. Similarly, numerical details for AUROC from Figure 11, accuracy, and f1-score are fully detailed in Appendix C.

4.3 Comparative Evaluation on Personalization Techniques

We conducted a thorough comparison of the overall results of the personalization techniques. Table 7 summarizes the performance of each personalized model under its optimal setting alongside the performance of the non-personalized models. Fine-tuning tended to perform better than the majority of cases. Out of 27 dataset-model combinations, fourteen showed that the fine-tuning model delivered the best performance. Following this, cluster-specific models were the most effective in six combinations, which were mostly from the ASCERTAIN dataset. Lastly, the hybrid model demonstrated the best performance in five combinations. However, there were instances where none of the personalization models significantly outperformed the non-personalized models, highlighting the complexity of personalization in affect recognition.

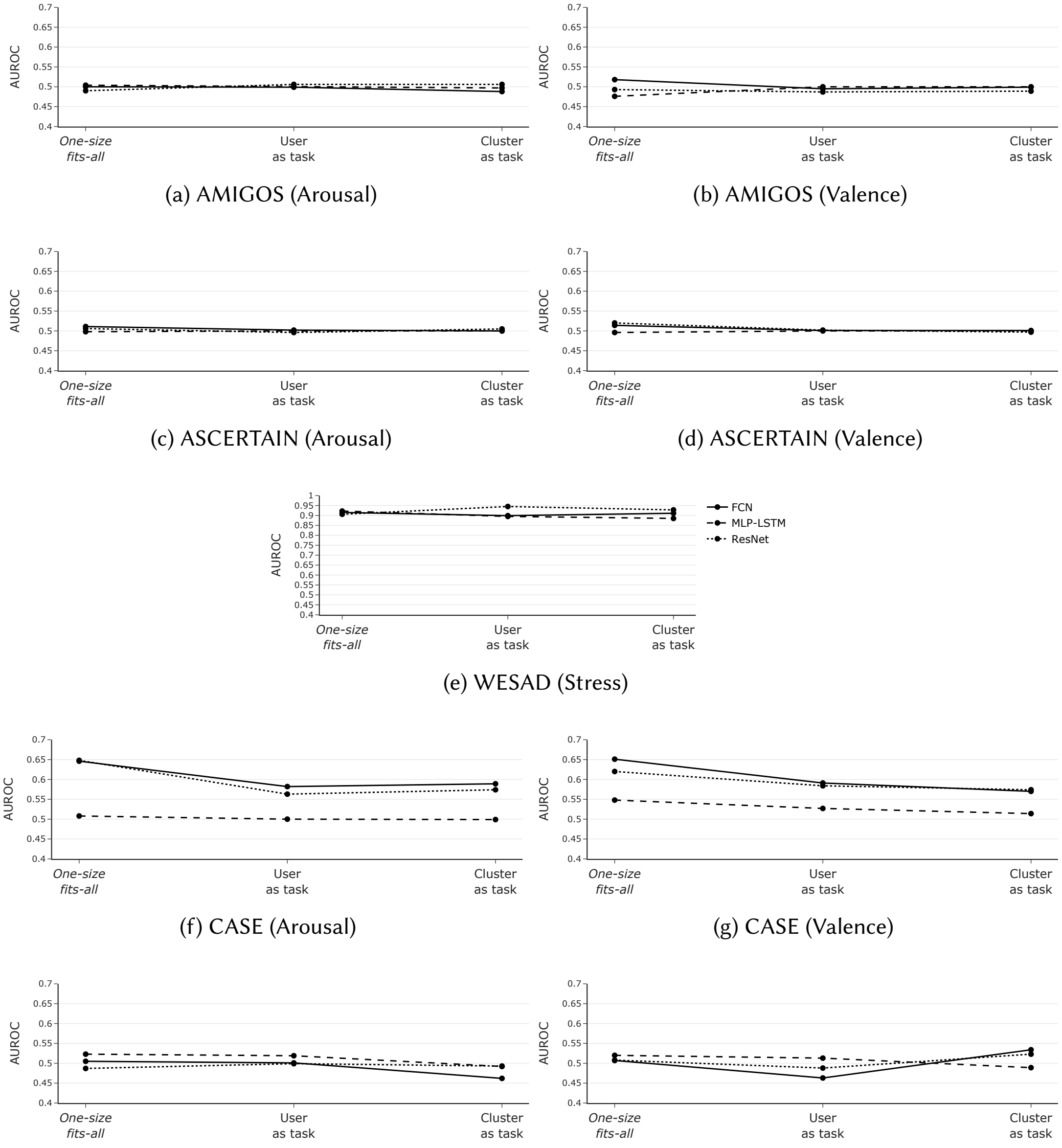
We then examined the statistical significance of performance differences between the non-personalized model and each personalized model at its optimal setting by conducting independent t-tests. We marked a marginal level of statistical significance ($p < 0.1$) with †, and a more stringent significance ($p < 0.05$) with * in Table 7. Due to the limited sample size and individual variations in the leave-one-participant-out cross-validation, statistical significance was not achieved across all combinations, reflecting the inherent challenges in personalization for affect recognition. Notably, our findings reveal that statistical significance was observed in only four combinations: two from fine-tuning, one from hybrid, and one from cluster-specific. Fine-tuning was not always statistically superior to non-personalized models in every combination. This indicates that while fine-tuning showed promise, it was not universally superior to non-personalized models in every combination. We discuss the implications of our findings in the following section, including the limitations and future work.

5 Discussion

We compare our findings with previous literature and discuss the challenges and future work in personalization for affective state recognition using wearable sensors.

5.1 Comparison of Current Findings with Existing Studies

Non-Personalized Model: The performance of our non-personalized models using the AMIGOS, ASCERTAIN, WESAD, CASE, and K-EmoCon datasets was found to be well aligned with the performances reported in existing studies. The AMIGOS and ASCERTAIN dataset resulted in an average AUROC of 50% and 51% in our evaluation, which mirrors the results of [14]. Our best architecture for stress classification with the WESAD dataset yielded an f1-score of 87% and an average accuracy of 85%. The results are slightly lower than those reported in prior studies [14, 67], but the difference may be due to dataset protocol selection for evaluation. The CASE dataset showed an average accuracy of 54% and an f1-score of 42%, closely aligning with the baseline performance for arousal detection by Zhang et al. [91]. For the K-EmoCon dataset, our models achieved an average accuracy of 52% and an f1-score of 45%. Yang et al. [84] reported higher f1-scores of 75% and 72% with BiLSTM and Transformer models, respectively, but they used 5-fold cross-validation, contrasting with our LOPO evaluation approach.



(a) AMIGOS (Arousal)

(b) AMIGOS (Valence)

(c) ASCERTAIN (Arousal)

(d) ASCERTAIN (Valence)

(e) WESAD (Stress)

(f) CASE (Arousal)

(g) CASE (Valence)

(h) K-EmoCon (Arousal)

(i) K-EmoCon (Valence)

Fig. 11. Multi-task Learning: AUROC across Various Settings. For the vertical scale, we used a default range of 0.4 to 0.7, except for the WESAD dataset, where we employed a range of 0.4 to 1.0.

Table 7. AUROC Summary of Non-personalized and Personalized Models.

Dataset	Architecture	Personalization Techniques				
		Non-Personalized	Fine Tuning	Hybrid	Cluster Specific	Multi-task Learning
AMIGOS (Arousal)	FCN	0.500 (0.100)	0.505 (0.159)	**0.512 (0.122)**	0.506 (0.122)	0.499 (0.038)
	MLP-LSTM	0.504 (0.100)	**0.538 (0.140)**	0.476 (0.107)	0.514 (0.129)	0.500 (0.000)
	ResNet	0.490 (0.115)	**0.546 (0.147)**	0.518 (0.136)	0.521 (0.078)	0.506 (0.044)
AMIGOS (Valence)	FCN	0.518 (0.131)	0.502 (0.159)	0.494 (0.145)	**0.531 (0.125)**	0.499 (0.021)
	MLP-LSTM	0.476 (0.109)	**0.528 (0.132)**†	0.511 (0.142)	0.515 (0.134)	0.500 (0.000)
	ResNet	0.493 (0.106)	**0.546 (0.147)**	0.515 (0.112)	0.513 (0.124)	0.489 (0.058)
ASCERTAIN (Arousal)	FCN	0.511 (0.071)	**0.521 (0.078)**	0.508 (0.067)	0.517 (0.071)	0.502 (0.026)
	MLP-LSTM	0.498 (0.035)	0.513 (0.073)	0.491 (0.056)	**0.517 (0.071)**†	0.500 (0.000)
	ResNet	0.506 (0.070)	0.511 (0.075)	0.505 (0.085)	**0.521 (0.078)**	0.505 (0.028)
ASCERTAIN (Valence)	FCN	0.514 (0.060)	0.515(0.075)	0.505 (0.075)	**0.520 (0.073)**	0.501 (0.009)
	MLP-LSTM	0.496 (0.047)	0.495 (0.060)	0.499 (0.035)	**0.507 (0.073)**	0.500 (0.000)
	ResNet	**0.520 (0.064)**	0.512(0.079)	0.515 (0.066)	0.518(0.082)	0.502 (0.029)
WESAD	FCN	0.915 (0.203)	0.973 (0.089)	**0.976 (0.074)**	0.849 (0.303)	0.911 (0.199)
	MLP-LSTM	0.922 (0.195)	**0.983 (0.053)**	0.913 (0.212)	0.874 (0.266)	0.895 (0.222)
	ResNet	0.906 (0.196)	0.969 (0.076)	**0.979 (0.066)**	0.857 (0.308)	0.945 (0.120)
CASE (Arousal)	FCN	0.646 (0.165)	**0.709 (0.173)**	0.655 (0.197)	0.613 (0.159)	0.589 (0.150)
	MLP-LSTM	0.508 (0.069)	**0.532 (0.105)**	0.520 (0.106)	0.510 (0.100)	0.500 (0.021)
	ResNet	0.648 (0.155)	**0.695 (0.162)**	0.646 (0.168)	0.617 (0.150)	0.574 (0.142)
CASE (Valence)	FCN	0.651 (0.159)	**0.688 (0.203)**	0.655 (0.217)	0.649 (0.132)	0.591 (0.139)
	MLP-LSTM	**0.548 (0.134)**	0.494 (0.038)	0.506 (0.089)	0.543 (0.134)	0.527 (0.094)
	ResNet	0.620 (0.169)	**0.676 (0.176)**	0.651 (0.200)	0.633 (0.154)	0.584 (0.159)
K-EmoCon (Arousal)	FCN	0.505 (0.176)	**0.594 (0.188)**	0.509 (0.358)	0.528 (0.152)	0.501 (0.146)
	MLP-LSTM	0.523 (0.173)	**0.672 (0.369)**	0.650 (0.373)	0.522 (0.172)	0.519 (0.139)
	ResNet	0.487 (0.188)	**0.659 (0.215)***	0.594 (0.312)	0.501 (0.136)	0.499 (0.142)
K-EmoCon (Valence)	FCN	0.507 (0.147)	**0.546 (0.229)**	0.443 (0.202)	0.519 (0.174)	0.534 (0.158)
	MLP-LSTM	0.520 (0.174)	0.619 (0.232)	**0.752 (0.255)***	0.526 (0.154)	0.513 (0.120)
	ResNet	0.508 (0.130)	0.602 (0.295)	**0.643 (0.349)**	0.528 (0.119)	0.523 (0.130)

Personalized Model (Fine-Tuning and Hybrid): Overall, in various datasets, the personalization technique of fine-tuning a one-size-fits-all model or building a hybrid model using a portion of the target individual's data tended to show in performance improvement. However, the optimal amount of data to use and the extent of layers to tune for the best performance varied depending on the classification task, the datasets used, and the model architectures involved. Our findings coincide with previous studies on personalization through fine-tuning that also suggested its beneficial impact on model performance. For example, Kathan et al. [29] reported that tuning the last two layers improved MAE in depression prediction by 6.78 percentage points. Hybrid model building in which a certain fraction of target user data is used for model training improved model performance as in prior studies [46, 76]. Our results showed that fine-tuning was comparable to or marginally better than hybrid models. Hybrid modeling tends to use more personal data and capture overall patterns instead of individual patterns. Prior studies [6, 62] warned that there could be considerable individual heterogeneity in user behaviors in that individuals' emotions depend on their contexts (e.g., places, social settings, and activities) or idiosyncratic (individual and instance-dependent) digital fingerprints. Fine-tuning general models with an individual's data may better adapt to idiosyncratic digital fingerprints.

Personalized Model (Cluster-Specific): Our evaluation showed that cluster-specific personalization did not consistently show performance improvement across different datasets. The effectiveness of clustering in previous literature has also varied. Can et al. [9] showed that cluster-specific models improved stress detection accuracy by 3.92 percentage points. Kathan et al. [29] reported that gender-based cluster-specific models slightly improved MAE in depression prediction by 7.91 percentage points. Conversely, Tervonen et al. [77], using the WESAD dataset, found similar results to ours, where cluster-specific models showed comparable or slightly lower stress detection performance than non-personalized models, with an average accuracy decrease of 1.25 percentage points and an average F1-score increase of 0.25 percentage points. Overall, cluster-specific personalization, which involves building models only using data from individuals similar to the target, does not necessarily guarantee performance improvement. One possible explanation is that some clustering criteria (e.g., age and gender) failed to group users with similar affective responses to stimuli, although the use of personality traits showed minor improvements (less than 0.02); e.g., both arousal and valence in ASCERTAIN and only arousal in AMIGOS (FCN). Previous work that observed improvement using cluster-specific models, such as Can et al.'s work [9], used PSS scores from a pre-survey for clustering instead of demographics and personality. Further studies should be conducted to understand what types of personal profiles are effective for clustering in the future.

Personalized Model (MTL: Multi-task Learning): In our analysis, across five datasets, MTL personalization did not show significant performance improvement. This outcome contrasts with previous studies that documented performance enhancement. Saeed et al. [65] developed a personalized stress detection model using physiological data (a user-as-task MTL neural network model) and reported an average increase of 2.87 percent in AUROC values [65]. Yu et al. [86] conducted personalized wellbeing detection using physiological, behavioral, and contextual data along with user-as-task and cluster-as-task MTL CNN and LSTM models. They found that user-as-task models outperformed cluster-as-task models, with an average increase of 9.83 percentage points in f1-score values. Similarly, Taylor et al. [74], using the same data as previous studies, created various traditional and DL models for mood-as-task and cluster-as-task. While mood-as-task was ineffective, the cluster-as-task models showed an increase in AUROC values ranging from 11 to 21 percent. However, such improvements may have originated from the fact that these studies used a *user-dependent* approach, dividing each participant's data (including unseen users) into train and test sets. In contrast, we used a *user-independent* approach where we did not use the target participant's data in the training set at all. Recently, Li et al.[39] conducted a user-independent evaluation similar to ours, comparing MTL models with one-size-fits-all models. Unlike our findings, they showed that this approach significantly reduced MAE values in a user-as-task setting. One possible explanation is that their work used a large dataset with 239 users; clustering based on gender and personality traits could potentially identify groups of users who share similar states of well-being, such as mood and health conditions.

Table 8. AUROC Summary of Non-personalized and Personalized Models for K-EmoPhone dataset.

Dataset	Architecture	Personalization Techniques				
		Non-Personalized	Fine Tuning	Hybrid	Cluster-Specific	Multi-task Learning
K-EmoPhone (Stress)	FCN	**0.534 (0.087)**	0.508 (0.146)	0.526 (0.095)	0.531 (0.104)	0.500 (0.025)
	MLP-LSTM	0.493 (0.055)	**0.506 (0.154)**	0.479 (0.086)	0.505 (0.077)	0.493 (0.041)
	ResNet	0.515 (0.084)	0.511 (0.126)	**0.525 (0.098)**	0.530 (0.097)	0.499 (0.033)

5.2 Towards Personalized Affect Recognition in the Wild

Beyond the datasets collected in the lab setting, it is also important to consider how personalization works by using datasets collected in the wild setting. To address this, we expanded our evaluation of the *user-dependent model* by using the K-EmoPhone dataset [28], an in-the-wild open dataset for affect recognition. Among the few available open datasets [44, 81], K-EmoPhone was selected because it provides a sufficient amount of emotion-label data via ESM and comprises a substantial number of participants with wearable data (N=47) [28]. K-EmoPhone offers a rich variety of physiological and behavioral sensor data, including ACC data at 8Hz, EDA at 8Hz, TEMP at 1Hz, and heart rate at 1Hz. We used the default configuration with a time window of 15 minutes for each label and a 1-minute window size [89]. The mean number of labels per participant was 557.23 (SD = 129.36; range = 360–830). We conducted exactly the same process as in lab-setting datasets, ranging from preprocessing to evaluation. Table 8 presents the performance metrics. (Details are provided in Appendix D, and the code is also available in the repository.) The overall performance of non-personalized models was comparable with that in the original paper [28]. Similar to the results of the ASCERTAIN dataset, we could not find statistically significant performance improvements in fine-tuning or hybrid approaches in the current parameter configurations.

When compared to the datasets collected in the laboratory, there are several practical challenges in extending the current work to the in-the-wild setting. We set a fixed time window for aggregating the physiological and behavioral data, but the optimal setting may differ across participants in free-living conditions. Furthermore, there could be missing or low-quality sensor data collected in the wild (e.g., PPG signals are very susceptible to motion artifacts [49]). One critical aspect of the K-EmoPhone dataset is the reliance on the ESM for emotion labeling. The timing of these self-reported labels relative to the physiological data is crucial, as discrepancies could influence model accuracy. The challenges of capturing real-time affective states accurately in naturalistic settings must be considered, especially since self-reported emotions might not always align with physiological signals. This misalignment could hinder the model's ability to learn and predict affective states effectively. This kind of user-specific variability could further stem from individual differences in baseline physiological responses, behavioral patterns, or even subjective interpretations of emotional states, thereby influencing optimal parameter setting and data quality issues. Therefore, there should be follow-up studies on optimizing models using in-the-wild datasets (e.g., varying time window sizes and missing data handling strategies) and analyzing the impact of behavior differences (e.g., context variations) on personalized affect recognition. Furthermore, exploring additional contextual information (e.g., location, time of day, user activity) or user interaction data (e.g., app usage) might improve the accuracy of personalized affect recognition in the wild [26, 37, 51].

5.3 Advancing User-dependent Personalization for Affect Recognition

The need for models capable of performing well on unseen participants is becoming increasingly evident [74, 83]. Using multiple datasets, we evaluated whether well-known personalization techniques can effectively recognize the affect of participants with a limited amount of labeled data. We found that, except for fine-tuning, the other personalization techniques did not always improve performance. Fine-tuning as a *user-dependent model* involves using data from the target participant. However, the techniques that *do not use the target's data* in the model training process (i.e., creating *user-independent* personalized models), such as cluster-specific and multi-task learning, were not successful across multiple datasets. As illustrated earlier, this may be because the profile survey information provided by the open datasets we used was insufficient for identifying *similar* participants for cluster-specific and multi-task learning. Nonetheless, many prior studies observed improvements through personalization in a *user-dependent* approach, even in multitask learning [65, 74, 86]. Our findings suggest that a certain amount of target data is necessary for effective personalization, although increasing the data size may not always lead to performance improvement. In our study, fine-tuning required a substantial portion of the target participant's data (20–50%), which could place a heavy labeling burden on users [78]. Future research should explore ways to alleviate this burden by balancing model accuracy with the amount of user feedback needed [75].

5.4 Limitations and Future Work

Our work has several limitations and requires further research. One limitation relates to hyperparameter optimization because we used the settings reported in the existing literature. In this study, we did not explore optimal settings for parameters such as sliding window size and machine learning algorithm hyperparameters to avoid introducing confounding effects. This omission could impact model performance and personalization, as different parameter settings may interact in complex ways that influence the effectiveness of the model. Given this limitation, potential users of our methods are encouraged to experiment with various parameter settings to find the optimal configuration for their specific applications. Individual users may enhance personalization performance by tailoring the model to meet their needs better.

Expanding the range of affective states and their classes could be beneficial. While our evaluation focused on binary classifications of arousal, valence, and stress, future work should include a broader range of affect labels, such as basic emotions (e.g., anger and sadness) or mental health (e.g., depressive moods), to provide a more comprehensive understanding of personalization. Beyond simple binary classifications, future research could explore multi-dimensional labeling (i.e., considering both arousal and valence simultaneously) [14, 93]. Additionally, incorporating more datasets from free-living conditions with contextual information from mobile and wearable devices could enhance insights into affective states in real-world scenarios. Lastly, exploring alternative clustering methods might offer valuable insights. Examining approaches beyond the trait-based methods used in our study, such as sensor data-driven clustering [2], could improve the performance and validity of cluster-specific and multi-task learning models.

Our systematic evaluation of human data using leave-one-participant-out cross-validation yielded limited statistical significance, and thus, the robustness of our findings is constrained. This is largely because the sample size of existing open datasets is small, and there are considerable individual differences. It is important to note that prior studies using open datasets also faced similar challenges of interpersonal variation [92, 93]. To address these limitations, future research should focus on collecting and analyzing larger and more diverse datasets to improve the generalizability and statistical power of the findings. Moreover, systematically evaluating recent techniques such as meta-learning, domain adaptation, and domain generalization could help improve the model's ability to generalize across different populations and conditions. Exploring these approaches and integrating multi-modal data sources, such as physiological signals and contextual information, will be critical in refining the models and understanding their practical implications in real-world applications.

Realizing user-dependent personalization also requires user interface and feedback mechanisms. Further research is needed to explore how affect recognition systems can collect meaningful feedback from users and how users can interact with these systems to correct or adjust their self-reported affective states. This might involve developing user interfaces that are intuitive and responsive to user inputs regarding their emotional states or adaptive to a user's changing contexts [4, 57].

6 Conclusion

This work takes a step towards systematically evaluating well-known personalization techniques (fine-tuning, hybrid, cluster-specific, and multi-task learning) for affect recognition modeling. We utilized five open datasets with physiological and behavioral signals collected in controlled environments. We then built non-personalized and personalized models through end-to-end deep learning, comparing their performances. Our focus was on testing models on new users with limited data, considering both user-dependent and independent settings. Our evaluation revealed that, except for the fine-tuning, other personalization techniques did not consistently enhance performance. This shows the essential need for a certain amount of an unseen user's data to successfully personalize models, highlighting the potential of user-in-the-loop approaches for model personalization. This sets a vital foundation for more detailed research on this finding, which might potentially act as a stepping stone in assessing the effectiveness of personalization in real-world settings specifically for new users with limited data available for model training. Moreover, to ensure reproducibility, we integrated all methods and open-sourced the implementations, enabling researchers systematically evaluate personalization algorithms.

Acknowledgments

This research was supported by the National Research Foundation of Korea (NRF), funded by the Korean government (MSIT) (2022R1A2C2011536, 2022M3J6A1063021).

References

[1] Mojtaba Khomami Abadi, Ramanathan Subramanian, Seyed Mostafa Kia, Paolo Avesani, Ioannis Patras, and Nicu Sebe. 2015. DECAF: MEG-based multimodal database for decoding affective physiological responses. *IEEE Transactions on Affective Computing* 6, 3 (2015), 209–222.

[2] Daniel A Adler, Fei Wang, David C Mohr, and Tanzeem Choudhury. 2022. Machine learning for passive mental health symptom prediction: Generalization across different longitudinal mobile sensing studies. *Plos one* 17, 4 (2022), e0266516.

[3] Riccardo Albertoni, Sara Colantonio, Piotr Skrzypczyński, and Jerzy Stefanowski. 2023. Reproducibility of Machine Learning: Terminology, Recommendations and Open Issues. *arXiv preprint arXiv:2302.12691* (2023).

[4] Swarnali Banik, Sougata Sen, Snehanshu Saha, and Surjya Ghosh. 2024. Towards Reducing Continuous Emotion Annotation Effort During Video Consumption: A Physiological Response Profiling Approach. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 8, 3, Article 91 (Sept. 2024), 32 pages.

[5] Leah K Barr, Jeffrey H Kahn, and W Joel Schneider. 2008. Individual differences in emotion expression: Hierarchical structure and relations with psychological distress. *Journal of Social and Clinical Psychology* 27, 10 (2008), 1045–1077.

[6] Lisa Feldman Barrett and Ajay B. Satpute. 2019. Historical pitfalls and new directions in the neuroscience of emotion. *Neuroscience Letters* 693 (2019), 9–18. Functional Neuroimaging of the Emotional Brain.

[7] Behnam Behinaein, Anubhav Bhatti, Dirk Rodenburg, Paul Hungler, and Ali Etemad. 2021. A transformer architecture for stress detection from ecg. In *Proceedings of the 2021 ACM International Symposium on Wearable Computers*. 132–134.

[8] Patricia J Bota, Chen Wang, Ana LN Fred, and Hugo Plácido Da Silva. 2019. A review, current challenges, and future possibilities on emotion recognition using machine learning and physiological signals. *IEEE Access* 7 (2019), 140990–141020.

[9] Yekta Said Can, Niaz Chalabianloo, Deniz Ekiz, Javier Fernandez-Alvarez, Giuseppe Riva, and Cem Ersoy. 2020. Personal stress-level clustering and decision-level smoothing to enhance the performance of ambulatory stress detection with smartwatches. *IEEE Access* 8 (2020), 38146–38163.

[10] Rich Caruana. 1997. Multitask learning. *Machine learning* 28 (1997), 41–75.

[11] Youngjae Chang, Akhil Mathur, Anton Isopoussu, Junehwa Song, and Fahim Kawsar. 2020. A systematic study of unsupervised domain adaptation for robust human-activity recognition. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 4, 1 (2020), 1–30.

[12] Ruixuan Dai, Chenyang Lu, Linda Yun, Eric Lenze, Michael Avidan, and Thomas Kannampallil. 2021. Comparing stress prediction models using smartwatch physiological signals and participant self-reports. *Computer Methods and Programs in Biomedicine* 208 (2021), 106207.

[13] John J Dudley and Per Ola Kristensson. 2018. A review of user interface design for interactive machine learning. *ACM Transactions on Interactive Intelligent Systems (TiiS)* 8, 2 (2018), 1–37.

[14] Maciej Dzieżyc, Martin Gjoreski, Przemysław Kazienko, Stanisław Saganowski, and Matjaž Gams. 2020. Can we ditch feature engineering? end-to-end deep learning for affect recognition from physiological sensor data. *Sensors* 20, 22 (2020), 6535.

[15] Maria Egger, Matthias Ley, and Sten Hanke. 2019. Emotion recognition from physiological signal analysis: A review. *Electronic Notes in Theoretical Computer Science* 343 (2019), 35–55.

[16] Anna Ferrari, Daniela Micucci, Marco Mobilio, and Paolo Napoletano. 2020. On the Personalization of Classification Models for Human Activity Recognition. *IEEE Access* 8 (2020), 32066–32079. https://doi.org/10.1109/ACCESS.2020.2973425

[17] Rebecca A Ferrer and Wendy Berry Mendes. 2018. Emotion, health decision making, and health behaviour. , 16 pages.

[18] Tor T. Finseth, Michael C. Dorneich, Stephen Vardeman, Nir Keren, and Warren D. Franke. 2023. Real-Time Personalized Physiologically Based Stress Detection for Hazardous Operations. *IEEE Access* 11 (2023), 25431–25454.

[19] Shruti Gedam and Sanchita Paul. 2021. A review on mental stress detection using wearable sensors and machine learning techniques. *IEEE Access* 9 (2021), 84045–84066.

[20] Taesik Gong, Yeonsu Kim, Jinwoo Shin, and Sung-Ju Lee. 2019. Metasense: few-shot adaptation to untrained conditions in deep mobile sensing. In *Proceedings of the 17th Conference on Embedded Networked Sensor Systems*. 110–123.

[21] Odd Erik Gundersen and Sigbjørn Kjensmo. 2018. State of the art: Reproducibility in artificial intelligence. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 32.

[22] Stephan Hamann and Turhan Canli. 2004. Individual differences in emotion processing. *Current opinion in neurobiology* 14, 2 (2004), 233–238.

[23] Jennifer Healey and Rosalind Picard. 2002. Driver Stress Data. https://www.media.mit.edu/groups/affective-computing/data/

[24] Mohammad Hossin and Md Nasir Sulaiman. 2015. A review on evaluation metrics for data classification evaluations. *International journal of data mining & knowledge management process* 5, 2 (2015), 1.

[25] Hassan Ismail Fawaz, Germain Forestier, Jonathan Weber, Lhassane Idoumghar, and Pierre-Alain Muller. 2019. Deep learning for time series classification: a review. *Data mining and knowledge discovery* 33, 4 (2019), 917–963.

[26] Gyuwon Jung, Sangjun Park, and Uichin Lee. 2024. DeepStress: Supporting Stressful Context Sensemaking in Personal Informatics Systems Using a Quasi-experimental Approach. In *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems* (Honolulu, HI, USA) *(CHI '24)*. Association for Computing Machinery, New York, NY, USA, Article 1000, 18 pages.

[27] Melih Kandemir, Akos Vetek, Mehmet Gönen, Arto Klami, and Samuel Kaski. 2014. Multi-task and multi-view learning of user state. *Neurocomputing* 139 (2014), 97–106.

[28] Soowon Kang, Woohyeok Choi, Cheul Young Park, Narae Cha, Auk Kim, Ahsan Habib Khandoker, Leontios Hadjileontiadis, Heepyung Kim, Yong Jeong, and Uichin Lee. 2023. K-emophone: A mobile and wearable dataset with in-situ emotion, stress, and attention labels. *Scientific data* 10, 1 (2023), 351.

[29] Alexander Kathan, Mathias Harrer, Ludwig Küster, Andreas Triantafyllopoulos, Xiangheng He, Manuel Milling, Maurice Gerczuk, Tianhao Yan, Srividya Tirunellai Rajamani, Elena Heber, et al. 2022. Personalised depression forecasting using mobile sensor data and ecological momentary assessment. *Frontiers in Digital Health* 4 (2022), 964582.

[30] Stamos Katsigiannis and Naeem Ramzan. 2017. DREAMER: A database for emotion recognition through EEG and ECG signals from wireless low-cost off-the-shelf devices. *IEEE journal of biomedical and health informatics* 22, 1 (2017), 98–107.

[31] Jonghwa Kim and Elisabeth André. 2008. Emotion recognition based on physiological changes in music listening. *IEEE transactions on pattern analysis and machine intelligence* 30, 12 (2008), 2067–2083.

[32] Zachary D King, Judith Moskowitz, Begum Egilmez, Shibo Zhang, Lida Zhang, Michael Bass, John Rogers, Roozbeh Ghaffari, Laurie Wakschlag, and Nabil Alshurafa. 2019. Micro-stress EMA: A passive sensing framework for detecting in-the-wild stress in pregnant mothers. *Proceedings of the ACM on interactive, mobile, wearable and ubiquitous technologies* 3, 3 (2019), 1–22.

[33] Sander Koelstra, Christian Muhl, Mohammad Soleymani, Jong-Seok Lee, Ashkan Yazdani, Touradj Ebrahimi, Thierry Pun, Anton Nijholt, and Ioannis Patras. 2011. Deap: A database for emotion analysis; using physiological signals. *IEEE transactions on affective computing* 3, 1 (2011), 18–31.

[34] Saskia Koldijk, Maya Sappelli, Suzan Verberne, Mark A Neerincx, and Wessel Kraaij. 2014. The swell knowledge work dataset for stress and user modeling research. In *Proceedings of the 16th international conference on multimodal interaction*. 291–298.

[35] Wouter M Kouw and Marco Loog. 2018. An introduction to domain adaptation and transfer learning. *arXiv preprint arXiv:1812.11806* (2018).

[36] Sylvia D Kreibig. 2010. Autonomic nervous system activity in emotion: A review. *Biological psychology* 84, 3 (2010), 394–421.

[37] Hansoo Lee, Auk Kim, SangWon Bae, and Uichin Lee. 2024. S-ADL: Exploring Smartphone-based Activities of Daily Living to Detect Blood Alcohol Concentration in a Controlled Environment. In *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems* (Honolulu, HI, USA) *(CHI '24)*. Association for Computing Machinery, New York, NY, USA, Article 1005, 25 pages.

[38] Uichin Lee, Kyungsik Han, Hyunsung Cho, Kyong-Mee Chung, Hwajung Hong, Sung-Ju Lee, Youngtae Noh, Sooyoung Park, and John M Carroll. 2019. Intelligent positive computing with mobile, wearable, and IoT devices: Literature review and research directions. *Ad Hoc Networks* 83 (2019), 8–24.

[39] Boning Li and Akane Sano. 2020. Extraction and interpretation of deep autoencoder-based temporal features from wearables for forecasting personalized mood, health, and stress. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 4, 2 (2020), 1–26.

[40] Da Li, Yongxin Yang, Yi-Zhe Song, and Timothy Hospedales. 2018. Learning to generalize: Meta-learning for domain generalization. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 32.

[41] Jialin Li, Alia Waleed, and Hanan Salam. 2023. A survey on personalized affective computing in human-machine interaction. *arXiv preprint arXiv:2304.00377* (2023).

[42] Xin Liu, Ziheng Jiang, Josh Fromm, Xuhai Xu, Shwetak Patel, and Daniel McDuff. 2021. MetaPhys: few-shot adaptation for non-contact physiological measurement. In *Proceedings of the conference on health, inference, and learning*. 154–163.

[43] M Maithri, U Raghavendra, Anjan Gudigar, Jyothi Samanth, Prabal Datta Barua, Murugappan Murugappan, Yashas Chakole, and U Rajendra Acharya. 2022. Automated emotion recognition: Current trends and future perspectives. *Computer methods and programs in biomedicine* 215 (2022), 106646.

[44] Stephen M Mattingly, Julie M Gregg, Pino Audia, Ayse Elvan Bayraktaroglu, Andrew T Campbell, Nitesh V Chawla, Vedant Das Swain, Munmun De Choudhury, Sidney K D'Mello, Anind K Dey, et al. 2019. The tesserae project: Large-scale, longitudinal, in situ, multimodal sensing of information workers. In *Extended Abstracts of the 2019 CHI Conference on Human Factors in Computing Systems*. 1–8.

[45] Matthew BA McDermott, Shirly Wang, Nikki Marinsek, Rajesh Ranganath, Luca Foschini, and Marzyeh Ghassemi. 2021. Reproducibility in machine learning for health research: Still a ways to go. *Science Translational Medicine* 13, 586 (2021), eabb1655.

[46] Lakmal Meegahapola, William Droz, Peter Kun, Amalia De Götzen, Chaitanya Nutakki, Shyam Diwakar, Salvador Ruiz Correa, Donglei Song, Hao Xu, Miriam Bidoglia, et al. 2023. Generalization and Personalization of Mobile Sensing-Based Mood Inference Models: An Analysis of College Students in Eight Countries. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 6, 4 (2023), 1–32.

[47] Lakmal Meegahapola, Hamza Hassoune, and Daniel Gatica-Perez. 2024. M3BAT: Unsupervised Domain Adaptation for Multimodal Mobile Sensing with Multi-Branch Adversarial Training. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous*

Technologies 8, 2 (2024), 1–30.

[48] Lakmal Meegahapola, Salvador Ruiz-Correa, Viridiana del Carmen Robledo-Valero, Emilio Ernesto Hernandez-Huerfano, Leonardo Alvarez-Rivera, Ronald Chenu-Abente, and Daniel Gatica-Perez. 2021. One more bite? Inferring food consumption level of college students using smartphone sensing and self-reports. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 5, 1 (2021), 1–28.

[49] Luca Menghini, Evelyn Gianfranchi, Nicola Cellini, Elisabetta Patron, Mariaelena Tagliabue, and Michela Sarlo. 2019. Stressing the Accuracy: Wrist-worn Wearable Sensor Validation over Different Conditions. *Psychophysiolpgy* 56, 11 (2019), e13441.

[50] Juan Abdon Miranda-Correa, Mojtaba Khomami Abadi, Nicu Sebe, and Ioannis Patras. 2018. Amigos: A dataset for affect, personality and mood research on individuals and groups. *IEEE Transactions on Affective Computing* 12, 2 (2018), 479–493.

[51] Varun Mishra, Tian Hao, Si Sun, Kimberly N. Walter, Marion J. Ball, Ching-Hua Chen, and Xinxin Zhu. 2018. Investigating the Role of Context in Perceived Stress Detection in the Wild. In *Proceedings of the 2018 ACM International Joint Conference and 2018 International Symposium on Pervasive and Ubiquitous Computing and Wearable Computers* (Singapore, Singapore) *(UbiComp '18)*. Association for Computing Machinery, New York, NY, USA, 1708–1716.

[52] Varun Mishra, Gunnar Pope, Sarah Lord, Stephanie Lewia, Byron Lowens, Kelly Caine, Sougata Sen, Ryan Halter, and David Kotz. 2020. Continuous detection of physiological stress with commodity hardware. *ACM transactions on computing for healthcare* 1, 2 (2020), 1–30.

[53] Varun Mishra, Sougata Sen, Grace Chen, Tian Hao, Jeffrey Rogers, Ching-Hua Chen, and David Kotz. 2020. Evaluating the reproducibility of physiological stress detection models. *Proceedings of the ACM on interactive, mobile, wearable and ubiquitous technologies* 4, 4 (2020), 1–29.

[54] Nilava Mukherjee, Sumitra Mukhopadhyay, and Rajarshi Gupta. 2022. Real-time mental stress detection technique using neural networks towards a wearable health monitor. *Measurement Science and Technology* 33, 4 (2022), 044003.

[55] Bahareh Nakisa, Mohammad Naim Rastgoo, Andry Rakotonirainy, Frederic Maire, and Vinod Chandran. 2020. Automatic emotion recognition using temporal multimodal deep learning. *IEEE Access* 8 (2020), 225463–225474.

[56] Randolph M Nesse. 1990. Evolutionary explanations of emotions. *Human nature* 1 (1990), 261–289.

[57] Sameer Neupane, Mithun Saha, Nasir Ali, Timothy Hnat, Shahin Alan Samiei, Anandatirtha Nandugudi, David M. Almeida, and Santosh Kumar. 2024. Momentary Stressor Logging and Reflective Visualizations: Implications for Stress Management with Wearables. In *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems* (Honolulu, HI, USA) *(CHI '24)*. Association for Computing Machinery, New York, NY, USA, Article 809, 19 pages.

[58] Cheul Young Park, Narae Cha, Soowon Kang, Auk Kim, Ahsan Habib Khandoker, Leontios Hadjileontiadis, Alice Oh, Yong Jeong, and Uichin Lee. 2020. K-EmoCon, a multimodal sensor dataset for continuous emotion recognition in naturalistic conversations. *Scientific Data* 7, 1 (2020), 293.

[59] Marco Perugini and Lisa Di Blas. 2002. Analyzing personality related adjectives from an eticemic perspective: the big five marker scales (BFMS) and the Italian AB5C taxonomy. *Big Five Assessment* (2002), 281–304.

[60] Rosalind W Picard. 2000. *Affective computing.* MIT press.

[61] Alessandro Pogliaghi, Elena Di Lascio, Shkurta Gashi, Emanuela Piciucco, Silvia Santini, and Martin Gjoreski. 2022. Multi-task Learning for Stress Recognition. In *Adjunct Proceedings of the 2022 ACM International Joint Conference on Pervasive and Ubiquitous Computing and the 2022 ACM International Symposium on Wearable Computers*. 202–206.

[62] Abhishek Pratap, David C. Atkins, Brenna N. Renn, Michael J. Tanana, Sean D. Mooney, Joaquin A. Anguera, and Patricia A. Areán. 2019. The accuracy of passive phone sensors in predicting daily mood. *Depression and Anxiety* 36, 1 (2019), 72–81.

[63] Juan Carlos Quiroz, Elena Geangu, and Min Hooi Yong. 2018. Emotion recognition using smart watch sensor data: Mixed-design study. *JMIR mental health* 5, 3 (2018), e10153.

[64] James A Russell. 1979. Affective space is bipolar. *Journal of personality and social psychology* 37, 3 (1979), 345.

[65] Aaqib Saeed, Tanir Ozcelebi, Johan Lukkien, Jan BF van Erp, and Stojan Trajanovski. 2018. Model adaptation and personalization for physiological stress detection. In *2018 IEEE 5th International Conference on Data Science and Advanced Analytics (DSAA)*. IEEE, 209–216.

[66] Akane Sano, Sara Taylor, Andrew W McHill, Andrew JK Phillips, Laura K Barger, Elizabeth Klerman, and Rosalind Picard. 2018. Identifying objective physiological markers and modifiable behaviors for self-reported stress and mental health status using wearable sensors and mobile phones: observational study. *Journal of medical Internet research* 20, 6 (2018), e210.

[67] Philip Schmidt, Attila Reiss, Robert Duerichen, Claus Marberger, and Kristof Van Laerhoven. 2018. Introducing wesad, a multimodal dataset for wearable stress and affect detection. In *Proceedings of the 20th ACM international conference on multimodal interaction*. 400–408.

[68] Johannes Schneider and Michalis Vlachos. 2021. Personalization of deep learning. In *Data Science–Analytics and Applications: Proceedings of the 3rd International Data Science Conference–iDSC2020*. Springer, 89–96.

[69] Karan Sharma, Claudio Castellini, Egon L van den Broek, Alin Albu-Schaeffer, and Friedhelm Schwenker. 2019. A dataset of continuous affect annotations and physiological signals for emotion analysis. *Scientific data* 6, 1 (2019), 196.

[70] Yuan Shi, Minh Hoai Nguyen, Patrick Blitz, Brian French, Scott Fisk, Fernando De la Torre, Asim Smailagic, Daniel P Siewiorek, Mustafa Al'Absi, Emre Ertin, et al. 2010. Personalized stress detection from physiological measurements. In *International symposium on quality of*

life technology. 28–29.

[71] Lin Shu, Jinyan Xie, Mingyue Yang, Ziyi Li, Zhenqi Li, Dan Liao, Xiangmin Xu, and Xinyi Yang. 2018. A review of emotion recognition using physiological signals. *Sensors* 18, 7 (2018), 2074.

[72] Ramanathan Subramanian, Julia Wache, Mojtaba Khomami Abadi, Radu L Vieriu, Stefan Winkler, and Nicu Sebe. 2016. ASCERTAIN: Emotion and personality recognition using commercial sensors. *IEEE Transactions on Affective Computing* 9, 2 (2016), 147–160.

[73] Salah Taamneh, Panagiotis Tsiamyrtzis, Malcolm Dcosta, Pradeep Buddharaju, Ashik Khatri, Michael Manser, Thomas Ferris, Robert Wunderlich, and Ioannis Pavlidis. 2017. A multimodal dataset for various forms of distracted driving. *Scientific data* 4, 1 (2017), 1–21.

[74] Sara Taylor, Natasha Jaques, Ehimwenma Nosakhare, Akane Sano, and Rosalind Picard. 2017. Personalized multitask learning for predicting tomorrow's mood, stress, and health. *IEEE Transactions on Affective Computing* 11, 2 (2017), 200–213.

[75] Ali Tazarv, Sina Labbaf, Amir Rahmani, Nikil Dutt, and Marco Levorato. 2023. Active reinforcement learning for personalized stress monitoring in everyday settings. In *Proceedings of the 8th ACM/IEEE International Conference on Connected Health: Applications, Systems and Engineering Technologies*. 44–55.

[76] Ali Tazarv, Sina Labbaf, Stephanie M Reich, Nikil Dutt, Amir M Rahmani, and Marco Levorato. 2021. Personalized stress monitoring using wearable sensors in everyday settings. In *2021 43rd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC)*. IEEE, 7332–7335.

[77] Jaakko Tervonen, Sampsa Puttonen, Mikko J Sillanpää, Leila Hopsu, Zsolt Homorodi, Janne Keränen, Janne Pajukanta, Antti Tolonen, Arttu Lämsä, and Jani Mäntyjärvi. 2020. Personalized mental stress detection with self-organizing map: From laboratory to the field. *Computers in Biology and Medicine* 124 (2020), 103935.

[78] Niels van Berkel, Denzil Ferreira, and Vassilis Kostakos. 2017. The Experience Sampling Method on Mobile Devices. *ACM Comput. Surv.* 50, 6, Article 93 (dec 2017), 40 pages.

[79] Gideon Vos, Kelly Trinh, Zoltan Sarnyai, and Mostafa Rahimi Azghadi. 2023. Generalizable machine learning for stress monitoring from wearable devices: a systematic literature review. *International Journal of Medical Informatics* (2023), 105026.

[80] Rui Wang, Min SH Aung, Saeed Abdullah, Rachel Brian, Andrew T Campbell, Tanzeem Choudhury, Marta Hauser, John Kane, Michael Merrill, Emily A Scherer, et al. 2016. CrossCheck: toward passive sensing and detection of mental health changes in people with schizophrenia. In *Proceedings of the 2016 ACM international joint conference on pervasive and ubiquitous computing*. 886–897.

[81] Rui Wang, Fanglin Chen, Zhenyu Chen, Tianxing Li, Gabriella Harari, Stefanie Tignor, Xia Zhou, Dror Ben-Zeev, and Andrew T Campbell. 2014. StudentLife: assessing mental health, academic performance and behavioral trends of college students using smartphones. In *Proceedings of the 2014 ACM international joint conference on pervasive and ubiquitous computing*. 3–14.

[82] Yan Wang, Wei Song, Wei Tao, Antonio Liotta, Dawei Yang, Xinlei Li, Shuyong Gao, Yixuan Sun, Weifeng Ge, Wei Zhang, et al. 2022. A systematic review on affective computing: Emotion models, databases, and recent advances. *Information Fusion* 83 (2022), 19–52.

[83] Xuhai Xu, Xin Liu, Han Zhang, Weichen Wang, Subigya Nepal, Yasaman Sefidgar, Woosuk Seo, Kevin S Kuehn, Jeremy F Huckins, Margaret E Morris, et al. 2023. GLOBEM: Cross-Dataset Generalization of Longitudinal Human Behavior Modeling. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 6, 4 (2023), 1–34.

[84] Kangning Yang, Benjamin Tag, Yue Gu, Chaofan Wang, Tilman Dingler, Greg Wadley, and Jorge Goncalves. 2022. Mobile emotion recognition via multiple physiological signals using convolution-augmented transformer. In *Proceedings of the 2022 International Conference on Multimedia Retrieval*. 562–570.

[85] Jason Yosinski, Jeff Clune, Yoshua Bengio, and Hod Lipson. 2014. How transferable are features in deep neural networks? *Advances in neural information processing systems* 27 (2014).

[86] Han Yu, Elizabeth B Klerman, Rosalind W Picard, and Akane Sano. 2019. Personalized wellbeing prediction using behavioral, physiological and weather data. In *2019 IEEE EMBS International Conference on Biomedical & Health Informatics (BHI)*. IEEE, 1–4.

[87] Han Yu and Akane Sano. 2020. Passive sensor data based future mood, health, and stress prediction: User adaptation using deep learning. In *2020 42nd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC)*. IEEE, 5884–5887.

[88] Alexandros Zenonos, Aftab Khan, Georgios Kalogridis, Stefanos Vatsikas, Tim Lewis, and Mahesh Sooriyabandara. 2016. HealthyOffice: Mood recognition at work using smartphones and wearable sensors. In *2016 IEEE International Conference on Pervasive Computing and Communication Workshops (PerCom Workshops)*. IEEE, 1–6.

[89] Panyu Zhang, Gyuwon Jung, Jumabek Alikhanov, Uzair Ahmed, and Uichin Lee. 2024. A Reproducible Stress Prediction Pipeline with Mobile Sensor Data. *Proceedings of the ACM on interactive, mobile, wearable and ubiquitous technologies* 8, 3 (2024).

[90] Tianyi Zhang, Abdallah El Ali, Alan Hanjalic, and Pablo Cesar. 2022. Few-shot learning for fine-grained emotion recognition using physiological signals. *IEEE Transactions on Multimedia* 25 (2022), 3773–3787.

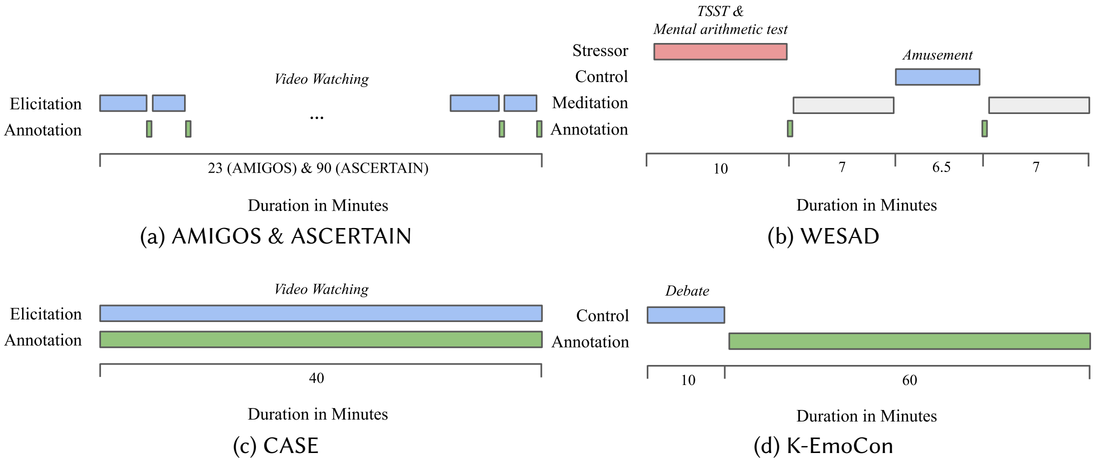
[91] Tianyi Zhang, Abdallah El Ali, Chen Wang, Alan Hanjalic, and Pablo Cesar. 2020. Cornet: Fine-grained emotion recognition for video watching using wearable physiological sensors. *Sensors* 21, 1 (2020), 52.

[92] Lili Zhu, Petros Spachos, Pai Chet Ng, Yuanhao Yu, Yang Wang, Konstantinos Plataniotis, and Dimitrios Hatzinakos. 2023. Stress detection through wrist-based electrodermal activity monitoring and machine learning. *IEEE Journal of Biomedical and Health Informatics* 27, 5 (2023), 2155–2165.

[93] M Sami Zitouni, Cheul Young Park, Uichin Lee, Leontios J Hadjileontiadis, and Ahsan Khandoker. 2022. LSTM-Modeling of Emotion Recognition Using Peripheral Physiological Signals in Naturalistic Conversations. *IEEE Journal of Biomedical and Health Informatics* 27, 2 (2022), 912–923.

A Appendix: Detailed Explanations for Each Dataset

The protocols for four datasets are depicted in Figure 12. In the following, we provide an overview of each dataset.



(a) AMIGOS & ASCERTAIN

(b) WESAD

(c) CASE

(d) K-EmoCon

Fig. 12. Timeline of the Lab Protocol for Each Dataset.

The **AMIGOS** and **ASCERTAIN** datasets are multimodal datasets collected while participants watched affective movie clips. In the AMIGOS dataset, participants engaged with 16 videos, while in the ASCERTAIN dataset, participants engaged with 36 videos. Both datasets include EEG, EDA, ECG, and ACC signal data. In AMIGOS, affective annotations were done for each video using a 9-point Likert scale, measuring arousal, valence, dominance, liking, and familiarity. Similarly, in ASCERTAIN, a 7-point Likert scale was used to measure arousal, valence, engagement, liking, and familiarity for each video. Additionally, both datasets include personality scores obtained through the Big Five personality test. The AMIGOS dataset also includes demographic information such as gender and age. But, we focused solely on the personality scores for clustering, in order to facilitate a comparison with the ASCERTAIN dataset.

The **WESAD** dataset is also a multimodal dataset designed to explore affect responses under controlled conditions with a specific study protocol. Participants engage in a series of activities, starting with a baseline phase (20 minutes; reading neutral material), followed by amusement (6.5 minutes; watching funny video clips), and stress (10 minutes; TSST public speaking and mental arithmetic task). Here, we only included the amusement and stress phases. Throughout all the activities, sensor data is collected from RespiBAN (ACC, RESP, ECG, EDA, EMG, TEMP) and E4 (BVP, EDA, TEMP, ACC). Self-report data was collected only after each condition using a modified Positive and Negative Affect Schedule (PANAS), State-Trait Anxiety Inventory (STAI), Self-Assessment Manikins (SAM), and Short Stress State Questionnaire (SSSQ) measures. Additionally, profile survey data includes demographic factors such as age and gender.

The **CASE** dataset is a multimodal dataset to explore real-time affect responses experienced during the viewing of diverse video content. Participants engaged with a total of 8 videos, each designed to evoke specific emotional states, with the entire session lasting approximately 21 minutes. Continuous affect annotation occurred concurrently with video-watching, facilitated by a joystick-based interface that allowed participants to report valence and arousal simultaneously on the X and Y axes, measured within the integer interval [-26225, 26225]. The dataset includes physiological recordings from ECG, BVP, EMG, EDA, RESP, and TEMP sensors. Furthermore, profile survey data captures demographic factors such as gender and age group, enhancing the dataset's contextual richness.

The **K-EmoCon** dataset is also a multimodal dataset to explore continuous affect responses experienced during naturalistic conversations. Participants conducted debates in pairs on a social issue, with the entire session lasting approximately 10 minutes. After the debate session, participants retrospectively annotated their affects at 5-second intervals while viewing the footage featuring themselves and their partners. The annotation involves a 5-point Likert scale for arousal and valence, a 4-point Likert scale for five different emotions indicative of stress state, and a choice among commonly used affective categories. Data from sensors, including NeuroSky MindWave Headset (EEG) and Empatica E4 (EDA, ACC, TEMP, BVP), were collected during the debate. Additionally, profile survey data includes demographics such as age and gender.

B Appendix: Details of Sampling Rates Before and After Downsampling

The initial sampling rate and the rate after downsampling for each dataset are detailed in Table 9.

Table 9. Sampling frequencies for each signal before (original sampling) and after downsampling

Dataset	Type of Signal	Original Sampling Rate	Processed Sampling Rate
AMIGOS	EEG	128Hz	8Hz
	EDA	128Hz	8Hz
	ACC	128Hz	8Hz
	ECG	256Hz	8Hz
ASCERTAIN	ECG	128Hz	64Hz
	EDA	128Hz	8Hz
	ACC	128Hz	8Hz
WESAD	Chest ECG	700Hz	70Hz
	Chest ACC	700Hz	10Hz
	Chest EMG	700Hz	10Hz
	Chest EDA	700Hz	7Hz
	Chest TEMP	700Hz	7Hz
	Chest Resp	700Hz	7Hz
	Wrist BVP	64Hz	64Hz
	Wrist ACC	32Hz	8Hz
	Wrist EDA	4Hz	4Hz
	Wrist TEMP	4Hz	4Hz
CASE	ECG	1000Hz	50Hz
	BVP	1000Hz	50Hz
	EDA	1000Hz	50Hz
	RESP	1000Hz	50Hz
	TEMP	1000Hz	50Hz
	EMG	1000Hz	50Hz
K-EmoCon	EDA	4Hz	4Hz
	ACC	32Hz	8Hz
	TEMP	4Hz	4Hz
	BVP	64Hz	64Hz

C Appendix: Detailed Performance Metrics of Personalized Models

Table 10, Table 11 and Table 12 present the performance outcomes of fine-tuned models across each dataset, considering variations in the number of layers tuned and the amount of data used for tuning.

Table 13 illustrates the performance results of cluster-specific models on each dataset, varying the number of clusters (K).

Table 14 outlines the comparative performance of multi-task learned models on various datasets, taking into account two distinct task definitions: user-as-task and cluster-as-task.

Table 10. Result of Analysis on Fine-Tuning Models (AMIGOS and ASCERTAIN (Arousal)). The abbreviations used are FC for FCN, ML MLP-LSTM, and Re for ResNet. For each dataset and model architecture pair, the highest AUROC value is highlighted.

	Layers	Data	**AMIGOS** (Arousal)			**AMIGOS** (Valence)			**ASCERTAIN** (Arousal)		
	Tuned	**Used**	**Accuracy**	**F1-score**	**AUROC**	**Accuracy**	**F1-score**	**AUROC**	**Accuracy**	**F1-score**	**AUROC**
FC	*All*	*20%*	0.477 (0.127)	0.428 (0.115)	0.486 (0.132)	0.498 (0.144)	0.349 (0.085)	0.500 (0.146)	0.516 (0.075)	0.392 (0.091)	0.508 (0.088)
		30%	0.473 (0.128)	0.428 (0.114)	0.489 (0.137)	0.496 (0.157)	0.346 (0.088)	0.492 (0.144)	0.517 (0.080)	0.392 (0.092)	0.507 (0.085)
		40%	0.497 (0.172)	0.345 (0.096)	0.490 (0.144)	0.497 (0.172)	0.345 (0.096)	0.490 (0.144)	0.515 (0.088)	0.389 (0.091)	0.509 (0.086)
		50%	0.499 (0.188)	0.342 (0.101)	0.502 (0.159)	0.499 (0.188)	0.342 (0.101)	**0.502 (0.159)**	0.518 (0.100)	0.389 (0.095)	0.508 (0.095)
	Last	*20%*	0.508 (0.131)	0.395 (0.103)	**0.505 (0.122)**	0.508 (0.118)	0.413 (0.100)	0.491 (0.134)	0.506 (0.072)	0.390 (0.085)	**0.516 (0.085)**
		30%	0.505 (0.139)	0.391 (0.102)	0.495 (0.123)	0.506 (0.122)	0.407 (0.097)	0.493 (0.133)	0.505 (0.077)	0.388 (0.084)	0.510 (0.085)
		40%	0.505 (0.150)	0.388 (0.105)	0.498 (0.126)	0.505 (0.132)	0.403 (0.099)	0.497 (0.133)	0.507 (0.083)	0.389 (0.087)	0.510 (0.086)
		50%	0.506 (0.163)	0.388 (0.108)	0.500 (0.133)	0.507 (0.147)	0.401 (0.107)	0.491 (0.151)	0.507 (0.092)	0.387 (0.089)	0.508 (0.098)
ML	*All*	*20%*	0.514 (0.131)	0.353 (0.076)	0.513 (0.067)	0.498 (0.146)	0.347 (0.099)	0.526 (0.135)	0.497 (0.075)	0.330 (0.034)	0.495 (0.052)
		30%	0.475 (0.098)	0.432 (0.083)	0.497 (0.104)	0.496 (0.157)	0.345 (0.100)	0.521 (0.135)	0.496 (0.082)	0.330 (0.037)	0.494 (0.054)
		40%	0.496 (0.172)	0.342 (0.105)	0.521 (0.137)	0.496 (0.172)	0.342 (0.105)	0.521 (0.137)	0.496 (0.092)	0.329 (0.041)	0.500 (0.056)
		50%	0.497 (0.190)	0.339 (0.114)	0.525 (0.161)	0.497 (0.190)	0.339 (0.114)	0.525 (0.161)	0.495 (0.013)	0.328 (0.047)	0.495 (0.061)
	Last	*20%*	0.469 (0.142)	0.328 (0.075)	0.520 (0.110)	0.503 (0.136)	0.357 (0.092)	0.514 (0.114)	0.504 (0.074)	0.334 (0.033)	0.511 (0.059)
		30%	0.462 (0.154)	0.320 (0.077)	0.518 (0.115)	0.502 (0.147)	0.355 (0.095)	0.522 (0.113)	0.504 (0.082)	0.334 (0.037)	0.506 (0.055)
		40%	0.459 (0.168)	0.316 (0.083)	0.526 (0.120)	0.503 (0.160)	0.354 (0.104)	0.525 (0.110)	0.504 (0.091)	0.334 (0.041)	0.506 (0.061)
		50%	0.456 (0.184)	0.313 (0.090)	**0.538 (0.140)**	0.502 (0.178)	0.353 (0.115)	**0.528 (0.132)**	0.505 (0.103)	0.334 (0.046)	**0.513 (0.073)**
Re	*All*	*20%*	0.476 (0.098)	0.436 (0.084)	0.494 (0.104)	0.451 (0.116)	0.359 (0.110)	0.534 (0.114)	0.508 (0.076)	0.397 (0.087)	0.506 (0.073)
		30%	0.514 (0.140)	0.352 (0.079)	0.514 (0.067)	0.445 (0.123)	0.353 (0.110)	0.538 (0.121)	0.508 (0.081)	0.396 (0.088)	0.506 (0.072)
		40%	0.439 (0.130)	0.345 (0.108)	0.537 (0.134)	0.439 (0.130)	0.345 (0.108)	0.537 (0.134)	0.508 (0.089)	0.394 (0.091)	**0.511 (0.075)**
		50%	0.434 (0.146)	0.341 (0.118)	**0.546 (0.147)**	0.434 (0.146)	0.341 (0.118)	**0.546 (0.147)**	0.506 (0.098)	0.390 (0.093)	0.508 (0.081)
	Last	*20%*	0.521 (0.125)	0.405 (0.101)	0.480 (0.121)	0.464 (0.137)	0.389 (0.134)	0.488 (0.136)	0.507 (0.076)	0.403 (0.092)	0.504 (0.096)
		30%	0.522 (0.138)	0.404 (0.105)	0.481 (0.132)	0.458 (0.144)	0.384 (0.135)	0.481 (0.140)	0.505 (0.082)	0.400 (0.093)	0.497 (0.091)
		40%	0.524 (0.150)	0.404 (0.111)	0.479 (0.142)	0.455 (0.154)	0.381 (0.139)	0.481 (0.149)	0.505 (0.089)	0.398 (0.094)	0.493 (0.091)
		50%	0.529 (0.163)	0.408 (0.117)	0.487 (0.148)	0.449 (0.170)	0.376 (0.150)	0.482 (0.164)	0.500 (0.101)	0.391 (0.095)	0.488 (0.105)

D Appendix: Detailed Performance Metrics of Models for K-EmoPhone dataset

Table 15 presents the detailed performance outcomes of non-personalized and hybrid models for K-EmoPhone. Also, it shows fine-tuned models, considering variations in the number of layers tuned and the amount of data used for tuning.

Table 11. Result of Analysis on Fine-Tuning Models (ASCERTAIN (Valence), WESAD and CASE (Arousal)). The abbreviations used are FC for FCN, ML MLP-LSTM, and Re for ResNet. For each dataset and model architecture pair, the highest AUROC value is highlighted.

	Layers Tuned	**Data Used**	**AMIGOS** (Arousal)			**AMIGOS** (Valence)			**ASCERTAIN** (Arousal)		
			Accuracy	**F1-score**	**AUROC**	**Accuracy**	**F1-score**	**AUROC**	**Accuracy**	**F1-score**	**AUROC**
FC	*All*	*20%*	0.540 (0.073)	0.420 (0.082)	0.511 (0.071)	0.919 (0.119)	0.873 (0.202)	**0.973 (0.089)**	0.574 (0.130)	0.484 (0.148)	0.664 (0.182)
		30%	0.540 (0.075)	0.418 (0.084)	**0.515 (0.075)**	0.928 (0.110)	0.879 (0.196)	0.971 (0.098)	0.574 (0.136)	0.485 (0.147)	0.666 (0.190)
		40%	0.539 (0.080)	0.412 (0.081)	0.513 (0.077)	0.877 (0.104)	0.834 (0.209)	0.922 (0.113)	0.574 (0.148)	0.482 (0.150)	0.656 (0.197)
		50%	0.542 (0.083)	0.413 (0.082)	0.510 (0.080)	0.948 (0.084)	0.890 (0.187)	0.965 (0.121)	0.578 (0.161)	0.478 (0.149)	0.659 (0.202)
	Last	*20%*	0.541 (0.073)	0.374 (0.064)	0.508 (0.054)	0.800 (0.214)	0.759 (0.261)	0.899 (0.237)	0.588 (0.121)	0.561 (0.130)	0.700 (0.174)
		30%	0.542 (0.075)	0.374 (0.065)	0.507 (0.058)	0.804 (0.217)	0.765 (0.262)	0.896 (0.252)	0.589 (0.133)	0.563 (0.141)	0.707 (0.171)
		40%	0.544 (0.078)	0.375 (0.065)	0.506 (0.061)	0.806 (0.272)	0.766 (0.306)	0.892 (0.265)	0.578 (0.143)	0.549 (0.151)	0.703 (0.172)
		50%	0.545 (0.081)	0.375 (0.065)	0.506 (0.061)	0.807 (0.231)	0.767 (0.272)	0.890 (0.273)	0.574 (0.148)	0.537 (0.151)	**0.709 (0.173)**
ML	*All*	*20%*	0.536 (0.079)	0.347 (0.034)	0.491 (0.064)	0.890 (0.203)	0.863 (0.248)	0.907 (0.210)	0.535 (0.106)	0.345 (0.045)	0.519 (0.081)
		30%	0.537 (0.082)	0.348 (0.035)	0.491 (0.067)	0.894 (0.207)	0.868 (0.249)	0.913 (0.202)	0.538 (0.118)	0.346 (0.050)	0.522 (0.103)
		40%	0.539 (0.085)	0.348 (0.036)	0.492 (0.072)	0.934 (0.215)	0.873 (0.253)	0.967 (0.196)	0.542 (0.132)	0.347 (0.056)	**0.532 (0.105)**
		50%	0.540 (0.088)	0.349 (0.037)	0.494 (0.074)	0.893 (0.229)	0.867 (0.262)	0.927 (0.188)	0.548 (0.150)	0.348 (0.063)	0.521 (0.097)
	Last	*20%*	0.538 (0.076)	0.352 (0.034)	**0.495 (0.060)**	0.902 (0.194)	0.874 (0.242)	0.974 (0.056)	0.547 (0.102)	0.351 (0.042)	0.521 (0.006)
		30%	0.537 (0.082)	0.348 (0.035)	0.491 (0.067)	0.903 (0.198)	0.875 (0.242)	0.977 (0.050)	0.551 (0.112)	0.352 (0.047)	0.520 (0.142)
		40%	0.539 (0.085)	0.348 (0.036)	0.492 (0.072)	0.904 (0.204)	0.876 (0.245)	0.979 (0.051)	0.557 (0.126)	0.354 (0.052)	0.516 (0.158)
		50%	0.540 (0.088)	0.349 (0.037)	0.494 (0.074)	0.904 (0.216)	0.877 (0.252)	**0.983 (0.053)**	0.565 (0.143)	0.356 (0.059)	0.526 (0.169)
Re	*All*	*20%*	0.532 (0.061)	0.437 (0.078)	0.508 (0.077)	0.827 (0.180)	0.756 (0.252)	0.924 (0.207)	0.547 (0.123)	0.477 (0.148)	0.668 (0.147)
		30%	0.533 (0.068)	0.431 (0.074)	0.501 (0.081)	0.900 (0.106)	0.844 (0.199)	**0.969 (0.076)**	0.549 (0.133)	0.477 (0.156)	0.679 (0.148)
		40%	0.537 (0.067)	0.438 (0.075)	**0.512 (0.079)**	0.894 (0.107)	0.867 (0.193)	0.917 (0.111)	0.541 (0.142)	0.467 (0.157)	0.670 (0.154)
		50%	0.540 (0.072)	0.444 (0.080)	0.510 (0.080)	0.909 (0.107)	0.856 (0.196)	0.957 (0.140)	0.534 (0.163)	0.460 (0.169)	0.663 (0.169)
	Last	*20%*	0.535 (0.065)	0.439 (0.079)	0.511 (0.071)	0.799 (0.257)	0.751 (0.298)	0.906 (0.267)	0.609 (0.131)	0.570 (0.140)	0.691 (0.168)
		30%	0.533 (0.068)	0.431 (0.074)	0.501 (0.081)	0.801 (0.264)	0.751 (0.302)	0.906 (0.266)	0.610 (0.127)	0.566 (0.134)	0.694 (0.163)
		40%	0.537 (0.067)	0.438 (0.075)	**0.512 (0.079)**	0.801 (0.272)	0.750 (0.306)	0.907 (0.265)	0.597 (0.138)	0.555 (0.140)	0.692 (0.162)
		50%	0.540 (0.072)	0.444 (0.080)	0.510 (0.080)	0.802 (0.278)	0.745 (0.308)	0.906 (0.263)	0.593 (0.143)	0.544 (0.134)	**0.695 (0.162)**

Table 12. Result of Analysis on Fine-Tuning Models (CASE (Arousal) and K-EmoCon).

	Layers Tuned	**Data Used**	**AMIGOS** (Arousal)			**AMIGOS** (Valence)			**ASCERTAIN** (Arousal)		
			Accuracy	**F1-score**	**AUROC**	**Accuracy**	**F1-score**	**AUROC**	**Accuracy**	**F1-score**	**AUROC**
FC	*All*	*20%*	0.623 (0.156)	0.562 (0.159)	0.665 (0.196)	0.486 (0.180)	0.400 (0.162)	0.498 (0.230)	0.466 (0.250)	0.339 (0.155)	0.468 (0.210)
		30%	0.612 (0.151)	0.544 (0.155)	0.675 (0.182)	0.493 (0.219)	0.388 (0.185)	0.490 (0.250)	0.453 (0.299)	0.295 (0.152)	0.449 (0.292)
		40%	0.621 (0.167)	0.545 (0.168)	0.684 (0.178)	0.519 (0.331)	0.397 (0.271)	0.568 (0.352)	0.466 (0.343)	0.308 (0.229)	0.426 (0.516)
		50%	0.629 (0.183)	0.539 (0.180)	**0.688 (0.203)**	0.461 (0.392)	0.351 (0.325)	0.518 (0.381)	0.475 (0.210)	0.374 (0.147)	0.482 (0.194)
	Last	*20%*	0.563 (0.144)	0.445 (0.148)	0.652 (0.193)	0.564 (0.170)	0.470 (0.151)	0.558 (0.191)	0.526 (0.224)	0.388 (0.124)	0.480 (0.214)
		30%	0.560 (0.154)	0.439 (0.149)	0.656 (0.197)	0.560 (0.193)	0.439 (0.156)	**0.594 (0.188)**	0.521 (0.258)	0.388 (0.197)	0.424 (0.219)
		40%	0.563 (0.144)	0.445 (0.148)	0.652 (0.193)	0.581 (0.258)	0.436 (0.207)	0.552 (0.240)	0.509 (0.281)	0.366 (0.218)	0.463 (0.242)
		50%	0.554 (0.188)	0.422 (0.154)	0.640 (0.218)	0.574 (0.315)	0.448 (0.300)	0.453 (0.372)	0.492 (0.307)	0.329 (0.219)	**0.546 (0.229)**
ML	*All*	*20%*	0.546 (0.138)	0.348 (0.062)	0.481 (0.048)	0.478 (0.234)	0.372 (0.194)	0.600 (0.223)	0.431 (0.347)	0.324 (0.288)	**0.619 (0.232)**
		30%	0.550 (0.152)	0.349 (0.069)	**0.494 (0.038)**	0.467 (0.270)	0.377 (0.251)	0.643 (0.249)	0.438 (0.399)	0.332 (0.336)	0.524 (0.345)
		40%	0.556 (0.170)	0.349 (0.077)	0.490 (0.040)	0.448 (0.314)	0.345 (0.265)	0.651 (0.268)	0.436 (0.446)	0.381 (0.428)	0.428 (0.496)
		50%	0.563 (0.194)	0.350 (0.089)	0.492 (0.036)	0.422 (0.354)	0.344 (0.334)	**0.672 (0.369)**	0.437 (0.295)	0.292 (0.161)	0.589 (0.229)
	Last	*20%*	0.516 (0.140)	0.330 (0.060)	0.471 (0.042)	0.591 (0.191)	0.419 (0.119)	0.575 (0.227)	0.446 (0.317)	0.320 (0.230)	0.592 (0.175)
		30%	0.520 (0.145)	0.320 (0.059)	0.461 (0.040)	0.611 (0.226)	0.463 (0.218)	0.593 (0.290)	0.445 (0.347)	0.333 (0.283)	0.582 (0.174)
		40%	0.513 (0.140)	0.311 (0.055)	0.451 (0.046)	0.643 (0.270)	0.452 (0.233)	0.617 (0.305)	0.442 (0.385)	0.317 (0.296)	0.603 (0.268)
		50%	0.510 (0.190)	0.320 (0.080)	0.470 (0.034)	0.663 (0.345)	0.522 (0.340)	0.400 (0.380)	0.450 (0.419)	0.374 (0.393)	0.487 (0.142)
Re	*All*	*20%*	0.564 (0.151)	0.501 (0.174)	0.661 (0.172)	0.478 (0.227)	0.371 (0.144)	0.556 (0.231)	0.419 (0.245)	0.311 (0.141)	0.455 (0.197)
		30%	0.564 (0.175)	0.505 (0.188)	0.672 (0.167)	0.480 (0.228)	0.355 (0.136)	0.570 (0.266)	0.355 (0.300)	0.240 (0.153)	0.338 (0.223)
		40%	0.588 (0.189)	0.528 (0.191)	**0.676 (0.176)**	0.471 (0.249)	0.378 (0.240)	0.521 (0.252)	0.336 (0.313)	0.219 (0.166)	0.601 (0.435)
		50%	0.597 (0.212)	0.542 (0.212)	0.672 (0.201)	0.447 (0.316)	0.337 (0.262)	0.604 (0.193)	0.448 (0.210)	0.345 (0.116)	0.467 (0.212)
	Last	*20%*	0.531 (0.159)	0.480 (0.171)	0.650 (0.170)	0.565 (0.157)	0.455 (0.142)	0.508 (0.198)	0.445 (0.177)	0.365 (0.134)	0.579 (0.231)
		30%	0.541 (0.161)	0.495 (0.178)	0.640 (0.170)	0.580 (0.169)	0.458 (0.156)	0.531 (0.204)	0.426 (0.196)	0.331 (0.133)	0.478 (0.202)
		40%	0.540 (0.146)	0.480 (0.160)	0.620 (0.166)	0.598 (0.201)	0.449 (0.161)	**0.659 (0.215)**	0.435 (0.221)	0.320 (0.138)	**0.602 (0.295)**
		50%	0.570 (0.210)	0.501 (0.201)	0.631 (0.190)	0.567 (0.243)	0.383 (0.153)	0.628 (0.193)	0.448 (0.282)	0.303 (0.162)	0.569 (0.329)

Table 13. Result of Analysis on Cluster Specific Models.

	Value of K	**AMIGOS** (Arousal)			**AMIGOS** (Valence)			**ASCERTAIN** (Arousal)		
		Accuracy	**F1-score**	**AUROC**	**Accuracy**	**F1-score**	**AUROC**	**Accuracy**	**F1-score**	**AUROC**
FC	*Silhouette score*	0.504 (0.134)	0.391 (0.113)	0.505 (0.130)	0.514 (0.104)	0.415 (0.099)	**0.531 (0.125)**	0.518 (0.064)	0.403 (0.087)	**0.517 (0.071)**
	Fixed (K=2)	0.503 (0.115)	0.372 0.087)	0.481 (0.124)	0.481 (0.122)	0.383 (0.106)	0.484 (0.133)	0.503 (0.062)	0.388 (0.084)	0.508 (0.071)
	Fixed (K=3)	0.503 (0.115)	0.372 (0.087)	0.481 (0.124)	0.521 (0.124)	0.386 (0.094)	0.511 (0.123)	0.487 (0.058)	0.363 (0.061)	0.506 (0.069)
	Fixed (K=4)	0.502 (0.111)	0.379 (0.094)	**0.506 (0.122)**	0.492 (0.116)	0.355 (0.078)	0.510 (0.112)	0.503 (0.064)	0.392 (0.081)	0.503 (0.072)
	Fixed (K=5)	0.491 (0.118)	0.381 (0.098)	0.504 (0.132)	0.480 (0.122)	0.365 (0.094)	0.484 (0.141)	0.518 (0.056)	0.401 (0.075)	0.509 (0.070)
ML	*Silhouette score*	0.505 (0.119)	0.36 2(0.079)	0.495 (0.130)	0.514 (0.123)	0.384 (0.109)	0.508 (0.119)	0.520 (0.063)	0.360 (0.064)	**0.517 (0.071)**
	Fixed (K=2)	0.493 (0.114)	0.359 (0.089)	0.483 (0.113)	0.500 (0.138)	0.364 (0.100)	**0.515 (0.134)**	0.500 (0.059)	0.348 (0.045)	0.502 (0.065)
	Fixed (K=3)	0.493 (0.114)	0.359 (0.089)	0.483 (0.113)	0.523 (0.109)	0.384 (0.093)	0.485 (0.108)	0.501 (0.064)	0.341 (0.041)	0.504 (0.073)
	Fixed (K=4)	0.507 (0.124)	0.375 (0.101)	0.461 (0.107)	0.505 (0.127)	0.360 (0.090)	0.493 (0.106)	0.504 (0.065)	0.347 (0.050)	0.504 (0.075)
	Fixed (K=5)	0.471 (0.142)	0.349 (0.111)	**0.514 (0.129)**	0.498 (0.133)	0.357 (0.095)	0.508 (0.136)	0.500 (0.068)	0.359 (0.072)	0.500 (0.067)
Re	*Silhouette score*	0.506 (0.126)	0.392 (0.106)	**0.491 (0.111)**	0.518 (0.124)	0.406 (0.095)	0.504 (0.130)	0.510 (0.062)	0.411 (0.088)	0.519 (0.086)
	Fixed (K=2)	0.512 (0.108)	0.382 (0.103)	0.464 (0.125)	0.494 (0.114)	0.398 (0.101)	0.502 (0.122)	0.512 (0.064)	0.404 (0.082)	0.499 (0.079)
	Fixed (K=3)	0.512 (0.108)	0.382 (0.103)	0.464 (0.125)	0.512 (0.114)	0.401 (0.104)	**0.513 (0.124)**	0.506 (0.064)	0.408 (0.083)	0.510 (0.072)
	Fixed (K=4)	0.488 (0.118)	0.389 (0.113)	0.482 (0.113)	0.491 (0.110)	0.384 (0.098)	0.497 (0.119)	0.511 (0.066)	0.409 (0.090)	0.491 (0.069)
	Fixed (K=5)	0.526 (0.111)	0.406 (0.087)	0.475 (0.122)	0.523 (0.104)	0.391 (0.072)	0.481 (0.103)	0.500 (0.058)	0.395 (0.085)	**0.521 (0.078)**

	Value of K	**ASCERTAIN** (Valence)			**WESAD**			**CASE** (Arousal)		
		Accuracy	**F1-score**	**AUROC**	**Accuracy**	**F1-score**	**AUROC**	**Accuracy**	**F1-score**	**AUROC**
FC	*Silhouette score*	0.525 (0.072)	0.390 (0.076)	0.512 (0.081)	0.767 (0.221)	0.676 (0.299)	0.801 (0.348)	0.522 (0.115)	0.430 (0.133)	0.606 (0.172)
	Fixed (K=2)	0.535 (0.070)	0.387 (0.068)	**0.520 (0.073)**	0.796 (0.254)	0.760 (0.289)	**0.849 (0.303)**	0.527 (0.114)	0.452 (0.141)	0.607 (0.166)
	Fixed (K=3)	0.531 (0.075)	0.392 (0.082)	0.519 (0.069)	0.711 (0.242)	0.626 (0.303)	0.772 (0.347)	0.531 (0.103)	0.442 (0.120)	**0.613 (0.159)**
	Fixed (K=4)	0.529 (0.076)	0.391 (0.075)	0.502 (0.061)	0.795 (0.213)	0.718 (0.292)	0.805 (0.319)	0.530 (0.110)	0.425 (0.127)	0.609 (0.149)
	Fixed (K=5)	0.537 (0.068)	0.401 (0.079)	0.517 (0.068)	0.740 (0.258)	0.654 (0.313)	0.766 (0.376)	0.531 (0.109)	0.413 (0.120)	0.604 (0.161)
ML	*Silhouette score*	0.533 (0.074)	0.349 (0.031)	0.496 (0.057)	0.712 (0.260)	0.630 (0.313)	0.792 (0.341)	0.506 (0.094)	0.333 (0.043)	**0.510 (0.100)**
	Fixed (K=2)	0.539 (0.068)	0.357 (0.037)	0.498 (0.057)	0.781 (0.263)	0.744 (0.298)	**0.874 (0.266)**	0.514 (0.093)	0.337 (0.042)	0.504 (0.113)
	Fixed (K=3)	0.541 (0.068)	0.355 (0.041)	0.491 (0.067)	0.777 (0.221)	0.713 (0.286)	0.801 (0.312)	0.509 (0.094)	0.335 (0.043)	0.502 (0.092)
	Fixed (K=4)	0.535 (0.070)	0.357 (0.050)	0.498 (0.065)	0.717 (0.196)	0.609 (0.266)	0.835 (0.288)	0.518 (0.091)	0.339 (0.041)	**0.510 (0.124)**
	Fixed (K=5)	0.533 (0.074)	0.360 (0.056)	**0.507 (0.073)**	0.770 (0.268)	0.710 (0.320)	0.780 (0.375)	0.529 (0.089)	0.344 (0.039)	**0.510 (0.138)**
Re	*Silhouette score*	0.516 (0.070)	0.411 (0.088)	0.517 (0.065)	0.710 (0.270)	0.637 (0.314)	0.804 (0.345)	0.516 (0.110)	0.424 (0.127)	0.587 (0.167)
	Fixed (K=2)	0.529 (0.073)	0.415 (0.083)	**0.518 (0.082)**	0.805 (0.227)	0.761 (0.270)	**0.857 (0.308)**	0.525 (0.106)	0.444 (0.125)	0.611 (0.162)
	Fixed (K=3)	0.522 (0.075)	0.408 (0.081)	0.509 (0.062)	0.723 (0.229)	0.660 (0.277)	0.809 (0.295)	0.519 (0.105)	0.425 (0.124)	0.615 (0.143)
	Fixed (K=4)	0.530 (0.070)	0.407 (0.080)	0.517 (0.074)	0.708 (0.231)	0.614 (0.290)	0.725 (0.356)	0.536 (0.118)	0.440 (0.138)	0.608 (0.155)
	Fixed (K=5)	0.505 (0.074)	0.393 (0.074)	0.501 (0.076)	0.681 (0.258)	0.587 (0.305)	0.708 (0.365)	0.549 (0.120)	0.445 (0.144)	**0.617 (0.150)**

	Value of K	**CASE** (Valence)			**K-EmoCon** (Arousal)			**K-EmoCon** (Valence)		
		Accuracy	**F1-score**	**AUROC**	**Accuracy**	**F1-score**	**AUROC**	**Accuracy**	**F1-score**	**AUROC**
FC	*Silhouette score*	0.525(0.125)	0.442(0.142)	0.616(0.155)	0.480 (0.121)	0.413 (0.111)	0.494 (0.149)	0.454 (0.183)	0.350 (0.117)	0.432 (0.139)
	Fixed (K=2)	0.576(0.113)	0.464(0.129)	0.631(0.155)	0.486 (0.127)	0.424 (0.120)	0.511 (0.143)	0.491 (0.191)	0.416 (0.176)	0.482 (0.111)
	Fixed (K=3)	0.564(0.118)	0.465(0.128)	**0.649(0.132)**	0.474 (0.117)	0.408 (0.115)	**0.528 (0.152)**	0.482 (0.144)	0.411 (0.125)	0.507 (0.147)
	Fixed (K=4)	0.532(0.132)	0.404(0.125)	0.613(0.145)	0.462 (0.144)	0.376 (0.124)	0.489 (0.159)	0.448 (0.201)	0.349 (0.135)	0.501 (0.130)
	Fixed (K=5)	0.542(0.120)	0.429(0.131)	0.616(0.152)	0.476 (0.142)	0.392 (0.115)	0.517 (0.143)	0.489 (0.153)	0.383 (0.097)	**0.519 (0.174)**
ML	*Silhouette score*	0.540 (0.116)	0.354 (0.071)	0.524 (0.147)	0.480 (0.153)	0.389 (0.118)	0.495 (0.167)	0.505 (0.223)	0.370 (0.103)	0.426 (0.143)
	Fixed (K=2)	0.528 (0.119)	0.342 (0.053)	0.511 (0.092)	0.467 (0.149)	0.392 (0.126)	0.513 (0.173)	0.479 (0.208)	0.360 (0.130)	0.462 (0.153)
	Fixed (K=3)	0.531 (0.119)	0.343 (0.053)	**0.543 (0.134)**	0.475 (0.149)	0.391 (0.119)	**0.522 (0.172)**	0.466 (0.144)	0.387 (0.120)	**0.526 (0.154)**
	Fixed (K=4)	0.540 (0.116)	0.347 (0.051)	0.463 (0.137)	0.452 (0.140)	0.365 (0.109)	0.498 (0.151)	0.545 (0.234)	0.386 (0.133)	0.463 (0.172)
	Fixed (K=5)	0.520 (0.121)	0.338 (0.053)	0.540 (0.135)	0.468 (0.156)	0.363 (0.113)	0.496 (0.140)	0.462 (0.166)	0.360 (0.133)	0.472 (0.122)
Re	*Silhouette score*	0.575 (0.107)	0.471 (0.129)	0.608 (0.160)	0.497 (0.135)	0.425 (0.111)	0.497 (0.151)	0.468 (0.207)	0.369 (0.127)	0.477 (0.129)
	Fixed (K=2)	0.584 (0.127)	0.480 (0.141)	0.607 (0.148)	0.500 (0.134)	0.433 (0.115)	0.494 (0.151)	0.489 (0.199)	0.385 (0.124)	0.474 (0.154)
	Fixed (K=3)	0.534 (0.129)	0.460 (0.150)	**0.633 (0.154)**	0.498 (0.131)	0.423 (0.113)	0.492 (0.149)	0.489 (0.094)	0.422 (0.093)	**0.528 (0.119)**
	Fixed (K=4)	0.511(0.126)	0.403 (0.122)	0.618 (0.160)	0.440 (0.112)	0.378 (0.100)	0.485 (0.145)	0.463 (0.150)	0.386 (0.127)	0.490 (0.155)
	Fixed (K=5)	0.537(0.098)	0.421 (0.115)	0.575 (0.172)	0.470 (0.145)	0.386 (0.107)	**0.501 (0.136)**	0.464 (0.142)	0.368 (0.100)	0.452 (0.127)

Table 14. Result of Analysis on Multi-task Learning Models.

	Task Definition	**AMIGOS** (Arousal)			**AMIGOS** (Valence)			**ASCERTAIN** (Arousal)		
		Accuracy	**F1-score**	**AUROC**	**Accuracy**	**F1-score**	**AUROC**	**Accuracy**	**F1-score**	**AUROC**
FC	*User-as-task*	0.456 (0.113)	0.319 (0.068)	**0.499 (0.038)**	0.491 (0.125)	0.336 (0.062)	0.495 (0.040)	0.510 (0.062)	0.337 (0.028)	**0.502 (0.026)**
	Cluster-as-task	0.478 (0.118)	0.326 (0.061)	0.488 (0.051)	0.448 (0.113)	0.309 (0.055)	**0.499 (0.021)**	0.505 (0.063)	0.338 (0.034)	0.500 (0.028)
ML	*User-as-task*	0.444 (0.112)	0.304 (0.051)	**0.500 (0.000)**	0.469 (0.122)	0.315 (0.055)	**0.500 (0.000)**	0.509 (0.062)	0.336 (0.027)	**0.500 (0.000)**
	Cluster-as-task	0.482 (0.124)	0.321 (0.056)	0.497 (0.017)	0.453 (0.116)	0.308 (0.053)	**0.500 (0.000)**	0.510 (0.062)	0.336 (0.027)	**0.500 (0.000)**
Re	*User-as-task*	0.439 (0.110)	0.324 (0.083)	**0.506 (0.074)**	0.512 (0.126)	0.349 (0.067)	0.487 (0.050)	0.508 (0.064)	0.344 (0.034)	0.496 (0.025)
	Cluster-as-task	0.456 (0.110)	0.320 (0.059)	**0.506 (0.044)**	0.459 (0.121)	0.322 (0.077)	**0.489 (0.058)**	0.512 (0.063)	0.342 (0.034)	**0.505 (0.028)**

	Task Definition	**ASCERTAIN** (Valence)			**WESAD**			**CASE** (Arousal)		
		Accuracy	**F1-score**	**AUROC**	**Accuracy**	**F1-score**	**AUROC**	**Accuracy**	**F1-score**	**AUROC**
FC	*User-as-task*	0.545 (0.067)	0.354 (0.032)	**0.501 (0.010)**	0.808 (0.211)	0.769 (0.255)	0.899 (0.213)	0.541 (0.103)	0.441 (0.121)	0.582 (0.122)
	Cluster-as-task	0.544 (0.068)	0.351 (0.029)	**0.501 (0.009)**	0.790 (0.214)	0.733 (0.273)	**0.911 (0.199)**	0.521 (0.115)	0.447 (0.136)	**0.589 (0.150)**
ML	*User-as-task*	0.544 (0.068)	0.351 (0.029)	**0.500 (0.001)**	0.856 (0.185)	0.814 (0.245)	**0.895 (0.222)**	0.505 (0.094)	0.333 (0.042)	**0.500 (0.021)**
	Cluster-as-task	0.544 (0.068)	0.351 (0.029)	**0.500 (0.000)**	0.860 (0.178)	0.823 (0.230)	0.885 (0.273)	0.496 (0.093)	0.329 (0.042)	0.499 (0.014)
Re	*User-as-task*	0.542 (0.071)	0.352 (0.029)	**0.502 (0.029)**	0.857 (0.169)	0.830 (0.208)	**0.945 (0.120)**	0.506 (0.109)	0.395 (0.116)	0.563 (0.140)
	Cluster-as-task	0.543 (0.065)	0.356 (0.033)	0.497 (0.021)	0.853 (0.195)	0.815 (0.253)	0.928 (0.164)	0.520 (0.107)	0.415 (0.121)	**0.574 (0.142)**

	Task Definition	**CASE** (Valence)			**K-EmoCon** (Arousal)			**K-EmoCon** (Valence)		
		Accuracy	**F1-score**	**AUROC**	**Accuracy**	**F1-score**	**AUROC**	**Accuracy**	**F1-score**	**AUROC**
FC	*User-as-task*	0.573 (0.114)	0.483 (0.142)	**0.591 (0.139)**	0.504 (0.125)	0.405 (0.106)	**0.501 (0.146)**	0.483 (0.107)	0.399 (0.101)	0.463 (0.128)
	Cluster-as-task	0.568 (0.120)	0.445 (0.141)	0.570 (0.121)	0.471 (0.140)	0.379 (0.113)	0.462 (0.145)	0.492 (0.124)	0.395 (0.095)	**0.534 (0.158)**
ML	*User-as-task*	0.561 (0.106)	0.356 (0.045)	**0.527 (0.094)**	0.540 (0.179)	0.375 (0.107)	**0.519 (0.139)**	0.551 (0.169)	0.381 (0.099)	**0.513 (0.120)**
	Cluster-as-task	0.561 (0.106)	0.356 (0.045)	0.514 (0.063)	0.517 (0.178)	0.340 (0.084)	0.492 (0.107)	0.534 (0.170)	0.360 (0.082)	0.489 (0.119)
Re	*User-as-task*	0.550 (0.106)	0.435 (0.102)	**0.584 (0.159)**	0.483 (0.128)	0.394 (0.124)	**0.499 (0.142)**	0.475 (0.117)	0.385 (0.117)	0.488 (0.141)
	Cluster-as-task	0.542 (0.133)	0.413 (0.096)	0.574 (0.150)	0.506 (0.128)	0.388 (0.103)	0.493 (0.134)	0.455 (0.126)	0.374 (0.125)	**0.523 (0.130)**

Table 15. Results of Analysis on K-EmoPhone (Stress) dataset.

Model	**Architecture**	**Accuracy**	**F1-score**	**AUROC**
Non-Personalized	FCN	0.646 (0.166)	0.403 (0.073)	0.534 (0.087)
	MLP-LSTM	0.655 (0.171)	0.389 (0.067)	0.493 (0.055)
	ResNet	0.631 (0.162)	0.406 (0.071)	0.515 (0.084)
Hybrid	FCN	0.680 (0.196)	0.425 (0.121)	0.526 (0.095)
	MLP-LSTM	0.690 (0.206)	0.431 (0.168)	0.479 (0.086)
	ResNet	0.662 (0.181)	0.440 (0.099)	0.525 (0.098)
Multi-task Learning *User-as-task*	FCN	0.655 (0.172)	0.389 (0.067)	0.498 (0.024)
	MLP-LSTM	0.655 (0.172)	0.389 (0.067)	0.488 (0.070)
	ResNet	0.655 (0.172)	0.389 (0.067)	0.492 (0.029)
Multi-task Learning *Cluster-as-task*	FCN	0.655 (0.172)	0.389 (0.067)	0.500 (0.025)
	MLP-LSTM	0.655 (0.172)	0.389 (0.067)	0.493 (0.041)
	ResNet	0.655 (0.171)	0.389 (0.067)	0.499 (0.033)

	Value of K	**KEmoPhone** (Stress)		
		Accuracy	**F1-score**	**AUROC**
FC	*Silhouette score*	0.615 (0.086)	0.161 (0.086)	0.490 (0.096)
	Fixed (K=2)	0.640 (0.072)	0.158 (0.072)	0.518 (0.071)
	Fixed (K=3)	0.638 (0.075)	0.163 (0.075)	0.513 (0.094)
	Fixed (K=4)	0.607 (0.081)	0.161 (0.081)	0.517 (0.090)
	Fixed (K=5)	0.626 (0.089)	0.169 (0.089)	**0.531 (0.104)**
ML	*Silhouette score*	0.638 (0.076)	0.174 (0.076)	0.493 (0.079)
	Fixed (K=2)	0.625 (0.081)	0.193 (0.081)	**0.505 (0.077)**
	Fixed (K=3)	0.633 (0.074)	0.185 (0.074)	0.459 (0.095)
	Fixed (K=4)	0.632 (0.075)	0.179 (0.075)	0.497 (0.078)
	Fixed (K=5)	0.654 (0.070)	0.165 (0.070)	0.476 (0.091)
Re	*Silhouette score*	0.606 (0.076)	0.131 (0.076)	0.506 (0.112)
	Fixed (K=2)	0.632 (0.073)	0.145 (0.073)	0.513 (0.080)
	Fixed (K=3)	0.621 (0.064)	0.142 (0.064)	0.510 (0.090)
	Fixed (K=4)	0.583 (0.075)	0.152 (0.075)	0.511 (0.084)
	Fixed (K=5)	0.591 (0.092)	0.145 (0.092)	**0.530 (0.097)**

	Layers	**Data**	**K-EmoPhone** (Stress)		
	Tuned	**Used**	**Accuracy**	**F1-score**	**AUROC**
FC	*All*	*20%*	0.648 (0.176)	0.415 (0.120)	0.496 (0.120)
		30%	0.652 (0.177)	0.413 (0.117)	0.493 (0.129)
		40%	0.648 (0.177)	0.407 (0.116)	0.478 (0.133)
		50%	0.651 (0.174)	0.408 (0.116)	0.474 (0.154)
	Last	*20%*	0.606 (0.157)	0.421 (0.084)	0.503 (0.132)
		30%	0.606 (0.157)	0.420 (0.084)	0.503 (0.132)
		40%	0.613 (0.170)	0.426 (0.121)	0.503 (0.142)
		50%	0.624 (0.164)	0.432 (0.131)	**0.508 (0.146)**
ML	*All*	*20%*	0.656 (0.172)	0.391 (0.069)	0.505 (0.142)
		30%	0.658 (0.173)	0.391 (0.069)	**0.506 (0.154)**
		40%	0.656 (0.173)	0.391 (0.070)	0.494 (0.164)
		50%	0.658 (0.170)	0.392 (0.069)	0.502 (0.178)
	Last	*20%*	0.511 (0.153)	0.432 (0.119)	0.502 (0.154)
		30%	0.511 (0.153)	0.432 (0.119)	0.502 (0.154)
		40%	0.511 (0.162)	0.427 (0.126)	0.491 (0.154)
		50%	0.528 (0.147)	0.443 (0.122)	0.480 (0.145)
Re	*All*	*20%*	0.606 (0.151)	0.420 (0.090)	**0.511 (0.126)**
		30%	0.603 (0.156)	0.415 (0.096)	**0.511 (0.133)**
		40%	0.603 (0.156)	0.413 (0.101)	0.502 (0.141)
		50%	0.600 (0.158)	0.413 (0.105)	0.495 (0.148)
	Last	*20%*	0.606 (0.157)	0.421 (0.084)	0.503 (0.132)
		30%	0.511 (0.153)	0.432 (0.119)	0.502 (0.154)
		40%	0.511 (0.162)	0.427 (0.126)	0.491 (0.154)
		50%	0.528 (0.147)	0.443 (0.122)	0.480 (0.145)