Proceedings of Machine Learning Research 287:1–13, 2025 Conference on Health, Inference, and Learning (CHIL) 2025

Contrastive Pretraining for Stress Detection with Multimodal Wearable Sensor Data and Surveys

Zeyu Yang ZYANG.PUB@GMAIL.COM
Han Yu HY29@RICE.EDU
Akane Sano AKANE.SANO@RICE.EDU
Department of Electrical and Computer Engineering, Rice University

Abstract

Stress adversely affects mental and physical health and underscores the importance of early detection. Some studies have utilized physiological signals from wearable sensors and other information to monitor stress levels in daily life. Recent studies use self-supervised methods due to the high cost of collecting stress labels. However, self-supervised learning using both time series and tabular features such as demographics, traits, and contextual information has been understudied. Therefore, there is a need to further investigate how a model can be effectively trained with different granularity of multimodal data and a limited number of labels. In this study, we introduce a self-supervised multimodal learning approach for stress detection that combines time series and tabular features. Our proposed method presents a promising solution for effectively monitoring stress using multimodal data.

Data and Code Availability This study uses two datasets (LifeSnaps and PMData) that can be accessed online (Yfantidou et al., 2022; Thambawita et al., 2020). Code is available at https://github.com/comp-well-org/clip-stress-detection.

Institutional Review Board (IRB) This research did not require IRB approval.

1. Introduction

Measuring stress is a crucial task in healthcare and well-being monitoring due to its significant impact on individuals' physical and mental health (Schneiderman et al., 2005). Wearable sensors offer a non-invasive and continuous method for monitoring physiological and behavioral signals such as heart rate, step counts, and sleep. Alongside physiological and behavioral data, surveys, self-reported information or other sensor information (e.g. demographics, traits, context) can be gathered through mobile applications to supplement user characteristics. By integrating multimodal data from wearable sensors, surveys, and others, a comprehensive view of an individual's stress levels and wellbeing can be achieved (Can et al., 2019).

Previous research has explored using machine learning models to detect stress or emotional health from wearable sensor data (Garg et al., 2021; Bobade and Vani, 2020; Yu and Sano, 2023) or electronic health records (Garriga et al., 2022). Previous studies used deep neural networks to learn directly from raw time series data (Li and Liu, 2020; Fernández and Anishchenko, 2018). Given that self-reported stress labels are typically scarce and costly to gather, recent efforts have focused on developing self-supervised machine learning models (Ericsson et al., 2022) with unlabeled data (Rabbani and Khan, 2022; Sarkar and Etemad, 2020). Researchers have demonstrated promising outcomes by pretraining deep learning models using unlabeled biosignals, audio-visual or textual data, and subsequently fine-tuning them on labeled data for stress detection (Sharma et al., 2023; Islam and Washington, 2023; Raghu et al., 2023). Recent studies have also examined the joint modeling of multimodal data such as images, videos, physiological signals, and text for stress detection (Bara et al., 2020; Seo et al., 2022; Wu et al., 2023; Gupta et al., 2022). Combining additional information with time series sensor data could supplement user characteristics and enhance the performance of machine learning models in healthcare (Jha et al., 2022). Integrating wearable time series data with demographic, contextual, and trait data from surveys, clinical assessments or other sensors has helped detect stress, emotional health, or cardiovascular health conditions (Taylor et al., 2017; Yu et al., 2024).

Multimodal models typically require a large amount of labeled data for training, which is often lacking in health-related datasets. Among self-supervised learning, various techniques to train a model without relying on labeled data, contrastive learning (Chopra et al., 2005) that trains a model by ensuring that representations of semantically similar examples are closer in embedding space compared to those of unrelated examples, has become particularly popular, especially with methods like SimCLR (Chen et al., 2020), and BYOL (Grill et al., 2020). In computer vision and natural language processing, contrastive learning has proven effective in learning representations from image-text pairs, as demonstrated by contrastive language-image pretraining (CLIP) (Radford et al., 2021). The training for CLIP, which involved a vast amount of image-text pairs and extensive computational resources, resulted in high-quality learned image and text representations. However, research on self-supervised multimodal learning with time series and tabular data (e.g., demographics, context, trait, self-reported data) is currently limited. Exploring methods to fuse information with various sampling rates to enhance model robustness and representations requires further investigation.

To fill these gaps, our study aims to propose a contrastive pretraining method for stress detection using multimodal data, including wearable times series data and demographic and contextual data. Our approach trains a multimodal encoder with multimodal data, including (1) wearable time series, (2) tabular features such as demographics, traits (e.g., personality types), contextual information (e.g., location), and a limited number of self-reported stress labels and (3) text descriptions generated from (1) and (2) through contrastive pretraining using time series and tabular data ((1) + (2)) and text (3). Once the model is pretrained, we use the pretrained model and wearable, demographics, and contextual information for downstream tasks, in this study, stress detection. By leveraging the complementary information from time series and tabular features, our method aims to learn representations that capture the underlying patterns of stress. The structured text descriptions guide the pretraining process, promoting the alignment of representations across modalities. We evaluate our method on two multimodal public datasets: LifeSnaps (Yfantidou et al., 2022) and PMData (Thambawita et al., 2020).

To summarize, the primary contributions of our paper are:

- We propose a multimodal encoder that integrates time series and tabular features.
- We introduce a method to generate structured text descriptions of the data, employ pretrained text encoders to obtain text representations, and utilize contrastive pretraining to learn multimodal representations.
- We demonstrate the effectiveness of our method on stress detection tasks using two multimodal datasets, especially when labeled samples are scarce.

The remainder of the paper is structured as follows: Section 2 details the methods employed in our study. Section 3 outlines the experimental setup, and Section 4 presents the results. Finally, Section 5 explains the limitations and future work, and Section 6 concludes the paper and discusses avenues for future research.

2. Methods

In this section, we outline the methods employed in our study. We begin by introducing the contrastive pretraining method and then explain how we adapt it for multimodal data that includes time series and tabular features.

2.1. Contrastive Pretraining

Building on the success of multimodal contrastive pretraining by linking visual and text information, we adapt it to learn representations for improved stress detection. Detecting stress often requires the integration of diverse data sources to capture the complexity of stress responses, including time series physiological signals (e.g., heart rate, steps) and tabular data (e.g., demographics, context data, and self-reported data through ecological momentary assessments).

To accomplish this, we design a multimodal encoder to encode both time series and tabular features. We then generate structured prompts based on the statistical characteristics of the time series data and the content of the tabular features. Note that our time series and tabular data and structured prompts do NOT include labels of the days. A pretrained BERT model (Devlin et al., 2018) tokenizes the generated prompts and extracts their representations by averaging the token embeddings. The contrastive loss is calculated using the cosine similarity between the representations of the prompts and the multimodal sequences. We fine-tune the pretrained model either with linear probing or end-to-end fine-tuning for downstream tasks. Figure 1 shows our contrastive pretraining framework and multimodal contrastive pertaining used for downstream tasks.

2.1.1. MULTIMODAL ENCODER

Our model is designed for datasets that include multichannel time series and tabular features. The time series data may consist of physiological signals such as heart

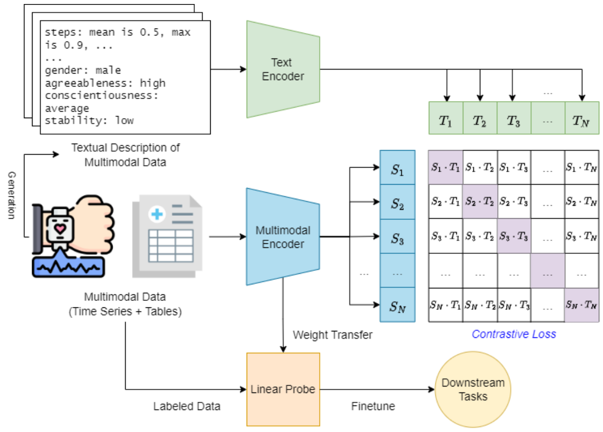

Figure 1: The contrastive pretraining framework generates textual descriptions from multimodal data, which includes time series and tabular features. A pretrained BERT model tokenizes the generated prompts and extracts their representations by averaging the token embeddings. The multimodal encoder is trained using contrastive methods and functions as a feature extractor. Following this, a linear probe is applied to allow for fine-tuning the model for downstream tasks.

rate and steps, capturing dynamic changes in the subject's state. Given the transformer's strength in handling sequential data like time series due to its ability to capture long-range dependencies, we utilize a transformer encoder for encoding the time series features. For the tabular features, which include questionnaire responses and demographic information, we employ a ResNet architecture that has proven effective in modeling tabular data (McElfresh et al., 2024). After obtaining representations for both time series and tabular features, we use a late fusion strategy to combine them. Specifically, we concatenate the representations and apply a linear layer to derive the multimodal representation. This late fusion strategy enables us to integrate information from both modalities at a high level and leverages the strengths of each representation.

2.1.2. PROMPT GENERATION

We generate structured text descriptions of the data to capture the statistical characteristics of time series channels, tabular feature content, and labels. For example, a prompt for the time series channel "steps" might be "steps: mean is 10.5, std is 1.1, min is 0.0, max is 24.0". Specifically, we employ 11 statistical features, including mean, standard deviation, minimum, maximum, 25% percentile, median, 75% percentile, root mean square, kurtosis, skewness (Groeneveld and Meeden, 1984), and interquartile range (IQR). Kurtosis is a descriptive statistic used to measure how data spreads between the center and the tails of a distribution. Skewness is a measure of the asymmetry of a distribution. IQR is a measure of statistical dispersion. In addition to time series, we generate descriptive sentences related to other modalities of information (e.g., demographics, traits, self-reports). For example, a prompt for gender might be "gender: female", and for the label, "label: stressed". Newline characters separate each descriptive sentence for a data modality.

The prompts are tokenized using a pretrained BERT model to obtain their representations, which are then averaged to form the final prompt representation. Significantly, our contrastive pretraining method can be applied to datasets with few labeled samples because the prompts are generated based on the data themselves and do not require labels. By utilizing descriptive text embeddings for contrastive tasks, we can leverage unlabeled data to pretrain the multimodal encoder that can then be fine-tuned on labeled data for downstream tasks. The main benefits of using text prompt embeddings compared to treating tables as numerical data are two fold. One, text embeddings can capture the semantic meaning of the data, while numerical data may not. Two, we can leverage pretrained text encoders to obtain high-quality representations of the prompts, which can be more effective than using numerical data directly.

2.1.3. PRETRAINING OBJECTIVE: CONTRASTIVE LOSS

The contrastive loss used in our approach is inspired by the Contrastive Language-Image Pretraining (CLIP) model (Radford et al., 2021), which aligns image and text representations by maximizing the similarity of matching pairs while minimizing the similarity of non-matching pairs. In our method, we adapt this loss to align multimodal data representations S_i with their corresponding structured prompts T_i. The cosine similarity of these paired embeddings is calculated as C_{ii}, and our objective function can be denoted as:

$$\mathcal{L} = \frac{1}{2N}\left(\sum_{i=1}^{N}\log\frac{\exp(C_{ii}/\tau)}{\sum_{j=1}^{N}\exp(C_{ij}/\tau)} + \sum_{i=1}^{N}\log\frac{\exp(C_{ii}/\tau)}{\sum_{j=1}^{N}\exp(C_{ji}/\tau)}\right) \tag{1}$$

Here, τ is a temperature parameter that controls the scaling of the similarities. The first term in the loss function corresponds to the similarity between each multimodal representation and its correct prompt representation relative to all other prompts in the batch. The second term does the same for each prompt representation relative to all other multimodal representations in the batch.

By minimizing this contrastive loss, we encourage the model to produce high cosine similarities for correct multimodal-prompt pairs and low cosine similarities for incorrect pairs. This helps the model learn aligned and discriminative representations for the multimodal data and their structured prompts.

2.2. Fine-tuning for Stress Detection

After pretraining the multimodal encoder with the contrastive loss, we fine-tune the model specifically for the task of stress detection. Fine-tuning involves two main approaches: linear probing and end-to-end fine-tuning.

For the linear probing, we freeze the pretrained multimodal encoder and train a linear classifier on top of the learned representations. Specifically, the representations obtained from the multimodal encoder are fed into a linear layer that outputs stress detection predictions. This approach evaluates the quality of the representations. Also, we perform end-to-end fine-tuning where the entire model, including the multimodal encoder, is fine-tuned on the labeled stress detection data. During this phase, the weights of the multimodal encoder are updated alongside the weights of the linear classifier. This fine-tuning process encourages the model to adapt the pretrained representations more specifically for the stress detection task.

3. Experiments

In this section, we detail the experimental setup and present the results of our study. We assess the performance of our proposed method using two public multimodal datasets: LifeSnaps and PMData.

3.1. Datasets and Data Preprocessing

3.1.1. LifeSnaps

LifeSnaps (Yfantidou et al., 2022) is a comprehensive multimodal dataset from 71 participants for more than 4 months. Of these participants, 42 are male and 29 are female, with half being under the age of 30 and the other half over 30. This dataset comprises three main types of data: ecological momentary assessments (EMAs), surveys, and physiological signals. Participants provide daily EMAs, including stress scores and whether participants are at home or not, as well as static information such as demographics and personality traits. Additionally, participants wear a smartwatch to record data on heart rate, steps taken, calories burned, distances traveled, and temperature. The prediction target is the stress level, which is categorized into below average, average, and above average based on the 33% and 66% percentiles among all user-days. In our study, we classify below average and average as negative (relaxed), and above average as positive (stressed). The biological signals collected by the smartwatch are processed into hourly features for each day. Along with survey data, these features are used to predict the stress level of a user on a given day. LifeSnaps includes data from 5850 days, with 228 of these being labeled: 80 days are classified as stressed and 148 days as relaxed.

3.1.2. PMData

PMData (Thambawita et al., 2020) is a sports logging dataset that captures lifelogging data, sports activities, and sleep patterns for 5 months from 16 persons. The participants range in age from 23 to 60, with three females and the remainder being males. It features hourly time series data, including heart rate, steps taken, calories burned, and distances traveled each day. The dataset also includes tabular data on demographics, sleep quality and duration, phone screen usage, sedentariness, activity levels, and injuries. Phone screen usage, sedentariness, and activity levels are measured in minutes. Labels in the PMData dataset indicate whether a user experienced stress on a given day (stressed vs relaxed). The dataset comprises in total of 2406 days, with 1727 labeled as relaxed, 352 labeled as stressed, and 327 remaining unlabeled.

3.1.3. Data Processing

We processed time series data from both datasets to extract hourly features. For the LifeSnaps dataset, these features include steps, heart rate, calories, distance, and temperature; for the PMData dataset, they include steps, heart rate, calories, and distance. In addition, we extracted tabular features from both datasets. LifeSnaps includes locations, personality traits, and demographics, while PMData incorporates sleep quality and duration, phone screen usage, sedentary behavior, activity levels, injuries, and demographics.

3.1.4. K-fold Cross-validation and Normalization

Due to the small dataset size, the train/test split could impact the results. To ensure a robust evaluation, we use

5-fold cross-validation. We also enforce constraints on label distribution to maintain the percentage of samples for each class. We compare two different cross-validation strategies: random splitting & non-random splitting. Random splitting allows participants to be mixed between the training and test sets. We apply constraints to the cross-validation process to ensure that the same participants are represented in both the training and test sets, except for those with fewer than five data points (two participants in the LifeSnaps dataset and one in the PMData dataset); these participants are included only in the training set. Non-random splitting keeps participants separate in the training and test sets by dividing them into two distinct groups. All data from each participant remains within either the training group or the test group, without mixing between them. In our experiments, we evaluated both strategies. Regarding the training setups, we can use either personalized or generalized models. Personalized models take user identities into account during testing, while generalized models do not. To clarify, the train/test split and training settings can be summarized as follows. We fill missing table values with -1 and time series values with 0.

- Random splitting with personalized models (with user ID).
- Random splitting with generalized models.
- Non-random splitting with generalized models.

The datasets include both time series channels and tabular features, with the primary key being a combination of user ID and date. In this study, we adopt global normalization strategies, which operate on the entire dataset. We handle time series channels and numerical tabular features separately. Quantile normalization is applied to both due to its robustness against outliers. Categorical tabular features are encoded using ordinal encoding.

3.2. Evaluation Metrics, Baselines, and Parameters

3.2.1. EVALUATION METRICS

We use the area under the receiver operating characteristic curve (AUC) as the primary evaluation metric for stress detection. AUC is a widely used metric for binary classification tasks that captures the trade-off between the true positive rate and the false positive rate across different decision thresholds. It is computed by plotting the true positive rate against the false positive rate at various thresholds and then calculating the area under the resulting curve.

3.2.2. BASELINES

To evaluate the effectiveness of our proposed method of contrastive language-sequence pretraining with multimodal data for stress detection, we compare it against several baseline approaches:

- Contrastive learning methods using only time series data. We specifically focus on three self-supervised learning methods: SimCLR (Chen et al., 2020), BYOL (Grill et al., 2020), and LEAVES (Yu et al., 2022). SimCLR and BYOL are two widely used contrastive learning methods that employ predefined data augmentation techniques to create augmented views of the data. In contrast, LEAVES makes data augmentation differentiable to enable the automatic learning of augmentation parameters for time series data.
- XGBoost (Chen and Guestrin, 2016) and random forest (Breiman, 2001) using tabular features only or tabular features and statistical features of time series data. The statistical features used are the same as those described in Section 2.1.2 for prompt generation.
- Supervised learning with raw time series (hourly data) and tabular features. We train multimodal deep neural networks on labeled data points of LifeSnaps and PMData.
- Using the average stress score of each participant from the training data to predict the stress score of the same participant in the test data if applicable, as suggested by (DeMasi et al., 2017).

For supervised methods, we utilize 100 percent of the labeled data points for training. For self-supervised methods, we use 100 percent of the unlabeled data points for pretraining, followed by fine-tuning the model with all labeled data points. XGBoost and random forest utilize statistical wearable and tabular features, while SimCLR and BYOL leverage raw time series data from wearable sensors. To ensure a fair comparison, we use the same neural network architecture for all deep learning baseline methods as we do for our proposed method.

3.2.3. PARAMETERS

We perform experiments for our proposed method and the baseline methods using 5-fold cross-validation and 4 random seeds. The supervised deep neural networks are trained for 300 epochs with a learning rate of 0.001 and a batch size of 512. The contrastive pretraining phase also lasts for 300 epochs, followed by fine-tuning for 150 epochs, both with a learning rate of 0.001 and a batch size of 512. For XGBoost and random forest, we use the default parameters as provided in the corresponding Python implementations. We use the implementation of transformer encoder with 3 layers and 4 attention heads from PyTorch. Our model is relatively small so it is computationally efficient. Running 1000 epochs including both training and inference takes around 150 seconds and occupies 1750 MB of GPU memory.

3.3. Interpretation of Contributing Features

We conduct experiments for interpreting our proposed deep learning models using the feature ablation algorithm provided by Captum (Kokhlikyan et al., 2020). The feature ablation algorithm is a permutation-based method that quantifies the impact of altering a feature on the prediction. Clinically, medical professionals could focus on these key features and conduct further analysis to enhance treatment or prevention strategies. In this paper, we focus on analyzing and interpreting random splitting with generalized models, as we believe this represents the most realistic scenario.

We analyze feature importance in the XGBoost model using SHAP (Lundberg and Lee, 2017) values, which quantify each feature's contribution to predictions. By ranking features based on their SHAP values, we identify the most influential factors in the model's decisions.

4. Results

4.1. Model Performance

4.1.1. RANDOM SPLITTING WITH GENERALIZED MODELS

Table 1 presents the test AUC of our proposed method alongside baseline models on the LifeSnaps and PMData datasets. For these experiments, we employed random splitting, where the same user can appear in both training and testing sets, with generalized models that do not incorporate user identity as a feature. After hyperparameter optimization for each fold of data split and each random seed, our method statistically outperformed XGBoost, random forest, SimCLR, and BYOL in terms of test AUC on both datasets. Compared to using XGBoost and random forest on survey data only and using deep learning methods on wearable data only, our self-supervised multimodal model has higher test AUC, which shows the effectiveness of using multimodal data for stress detection. LEAVES, using raw time series from wearable sensors as well, exhibited competitive performance on the LifeSnaps dataset, and the supervised method showed competitive performance on the PMData dataset.

Table 1: The values of test AUC in percentage on the LifeSnaps and PMData datasets for each method. The results are from 20 experiments with 5-fold cross-validation and 4 random seeds. The values are derived using all unlabeled data and fine-tuning with all labeled data for self-supervised models under random splitting with generalized models. Note that hyperparameter optimization is performed independently for each fold and each seed to obtain the best configuration.

	LifeSnaps	PMData
XGBoost (Survey Only)	64.76 ± 12.47	76.18 ± 2.85
Random Forest (Survey Only)	68.48 ± 10.13	73.09 ± 2.43
XGBoost (Multimodal)	62.78 ± 3.70	74.77 ± 1.79
Random Forest (Multimodal)	61.16 ± 6.61	72.38 ± 2.59
SimCLR (Wearable Only)	70.33 ± 4.31	73.99 ± 2.80
BYOL (Wearable Only)	68.55 ± 6.08	74.14 ± 2.45
LEAVES (Wearable Only)	**73.20 ± 5.65**	74.03 ± 1.81
Supervised (Multimodal)	67.18 ± 7.65	**80.81 ± 1.62**
Self-Supervised (Multimodal, Ours)	**72.45 ± 4.70**	**81.14 ± 1.79**

The higher standard deviations in the test AUC values for LifeSnaps are attributed to its smaller dataset size, totaling only 228 labels. Due to the limited number of labels, the supervised multimodal learning baseline achieved an average test AUC of approximately 67%. Through contrastive learning, our method improved performance by approximately 5%. LEAVES demonstrated comparable performance to our method, highlighting the significant role of time series modalities in stress detection within the LifeSnaps dataset.

In contrast, PMData showed smaller standard deviations in test AUC values due to its larger dataset size of 2079 labels. Both supervised multimodal learning and our method achieved over 80% test AUC, demonstrating substantial improvements through the integration of complementary information from tabular features compared to other methods. However, the enhancement in test AUC through contrastive learning was limited because PMData is almost fully labeled.

4.1.2. OTHER SPLITTING AND TRAINING STRATEGIES

Table 2 presents the performance of our model compared to the performance of the mean score baseline which uses the participant's mean stress score, across both datasets under different splitting and training strategies. These

Table 2: The values of test AUC in percentage on the LifeSnaps and PMData datasets with our proposed method trained on all unlabeled data and fine-tuned with all labeled data. P indicates personalized models through random mixing (RM) split with user ID as one of the features. G indicates generalized models with non-random mixing (NRM) user split.

	LifeSnaps	PMData
Ours (P, RM)	74.60 ± 4.97	89.02 ± 1.12
User Mean (P, RM)	80.37 ± 8.64	84.08 ± 2.12
Ours (G, NRM)	74.67 ± 9.92	89.36 ± 1.09

strategies include: (1) random splitting with personalized models, and (2) non-random splitting with generalized models. For the PMData dataset, in both the personalized and generalized settings, our proposed model after full fine-tuning significantly outperforms the baseline. However, in the LifeSnaps dataset, the mean score baseline method (using the participant's mean score) performed better than our proposed pre-training approach, though the high standard deviation suggests that this may be due to the limited number of labeled data points in the LifeSnaps dataset. An interesting phenomenon is that the performance of our proposed method in the generalized setting was not worse than the performance of the model in the personalized settings in the LifeSnaps and PMData datasets. The reason behind this result could be when the number of participants was small, and each participant had a unique combination of tabular features that could serve a "proxy" ID. This also indicates that stress might be more individual than generalizable. However, if we scale the model and training data, the generalized model could be more effective.

4.2. Contributing Features in the Proposed Model

In this section, we show the visualizations of feature importance for Fitbit channels and tabular features in the LifeSnaps and PMData datasets, as shown in Figure 2, 3, 4, and 5. The feature importance scores are based on the results for random splitting with generalized models.

Based on the feature ablation results, we found the following results:

- Steps and distances during late nights are more important than other Fitbit metrics, such as heart rate and calories, across both datasets. An increase in steps and distance during late nights could increase the likelihood of experiencing stress. This may be an indicator of poor or short sleep associated with higher stress.
- High BMI and low minutes in the most frequent location cluster measured by GPS are also related to stress in both datasets.
- Demographic factors such as old age and being female are associated with high stress.

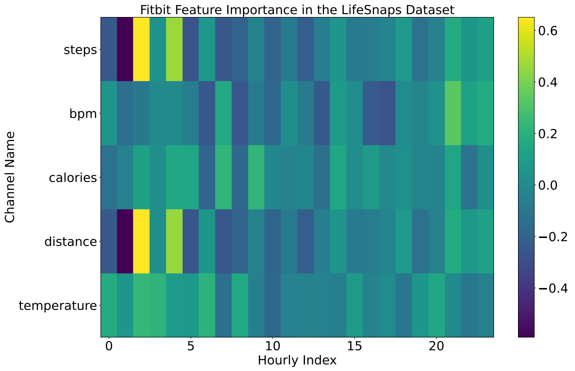

Figure 2: The heatmap of stress-related Fitbit channels in LifeSnaps.

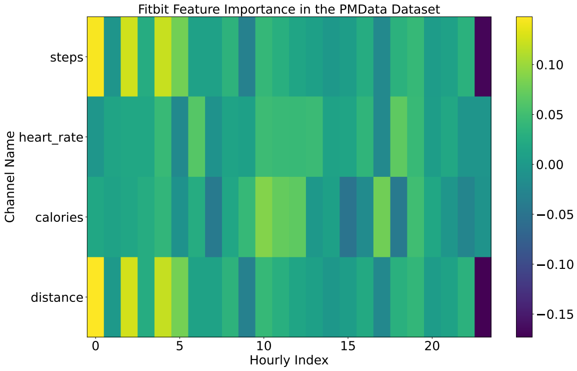

Figure 3: The heatmap of stress-related Fitbit channels in PMData.

In addition to these commonly observed factors in the two datasets, we also found other important features in each dataset. In the LifeSnaps dataset, among the top 20 contributing tabular features, high step goals and personality traits like low stability, low agreeableness, and low conscientiousness are linked to days with higher stress scores. Conversely, factors like not staying at home, transitions, work or school, working from home, gym visits, outdoor activities, entertainment, and high extraversion are associated with reduced stress.

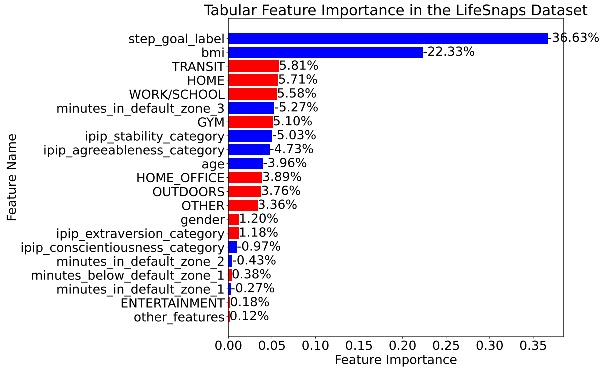

Figure 4: The top 20 stress-related tabular features in the LifeSnaps dataset. Negative numbers (blue bars) indicate that stress decreases if the feature values are replaced with zero, while positive numbers (red bars) suggest that stress increases if the feature values are replaced.

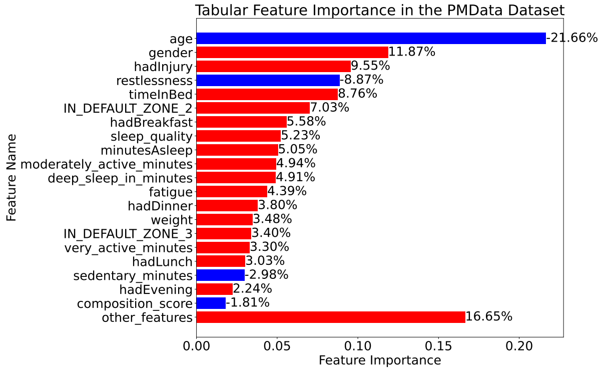

Figure 5: The top 20 stress-related tabular features in the PMData dataset. Negative numbers (blue bars) indicate that stress decreases if the feature value is replaced with zero, while positive numbers (red bars) suggest that stress increases if the features are replaced with zero.

In the PMData dataset, demographic factors are also linked to higher stress levels, with their impact being more pronounced compared to the LifeSnaps dataset. Days with elevated stress levels are further associated with short and poor-quality sleep, low active minutes, lower weight, and skipping meals like breakfast, lunch, or dinner. Interestingly, physical conditions like injury and fatigue seem to reduce the likelihood of stress, likely because, in the PMData, athletes tend to avoid workouts when they are injured or fatigued. Other factors contributing to stress include feeling restless, high body composition scores, and increased sedentary minutes.

4.3. Contributing Features in XGBoost

Figure 6 and Figure 7 show the top 20 features relevant to the stress labels in the XGBoost models with the LifeSnaps and PMData datasets.

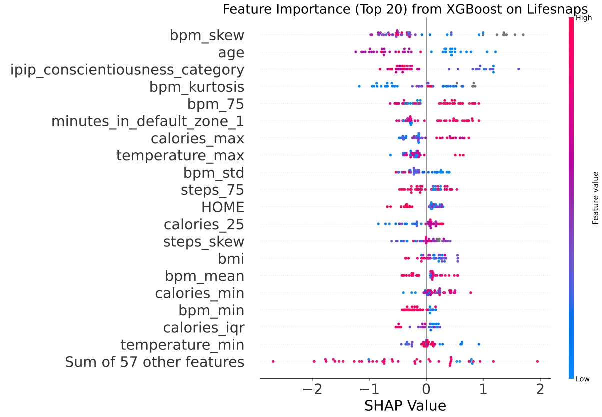

Figure 6: The top 20 stress-related features in LifeSnaps with XGBoost.

In the LifeSnaps dataset, among the top 20 contributing features, 11 are features computed from hourly wearable data and the remaining are tabular features. Statistical features of heart rate (5 features), temperature (2), and calories (4) significantly impact stress monitoring. Days with higher stress are associated with higher calorie consumption, higher temperature, and regular heart rate (lower heart rate kurtosis, low heart rate variability (HRV)) compared to days with relaxed labels. Among the tabular features, conscientiousness is the most relevant to stress, and higher conscientiousness is related to days of relaxed labels. Besides conscientiousness, factors such as not staying at home, younger age, higher body mass index, and lower gratification are also associated with stress.

In the PMData dataset, there are 6 wearable features and 14 tabular features among the top 20 contributing factors. The tabular features include sleep patterns (5 features), activity levels (4 features), food (2 features), and locations (3 features), all of which significantly influence stress levels. Days with higher stress are associated with less deep sleep and poor sleep quality and efficiency.

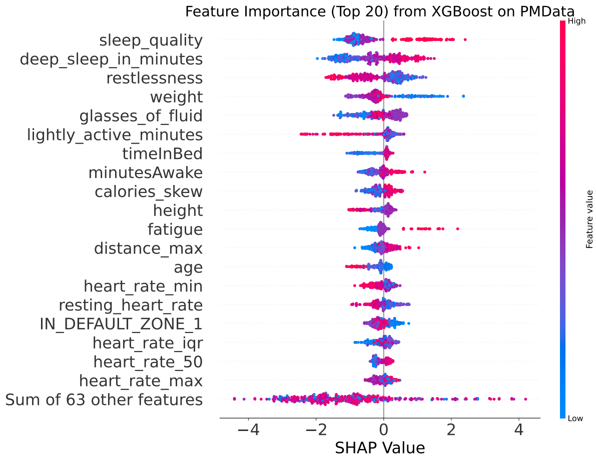

Figure 7: The top 20 stress-related features in PMData with XGBoost.

Additionally, higher restlessness is linked to stress, even though active minutes are higher on stressed days compared to relaxed ones. Higher weight and fluid intake are also associated with stress.

Compared to the LifeSnaps dataset, wearable hourly features have less impact on stress in the PMData dataset. Only 6 statistical features are among the top 20 contributors, including 4 heart rate features, 1 calorie feature, and 1 step feature. In PMData, stress is associated with a low and regular heart rate (low IQR and kurtosis), as well as high values of steps and calorie consumption.

Both datasets show the relationships between stress and low and regular heart rate values, higher restlessness, and lower conscientiousness. Previous studies (Kim and Dimsdale, 2007; Kim et al., 2018; Bartley and Roesch, 2011) have shown lower heart rate variability, poorer sleep quality, and lower conscientiousness in stress conditions. These findings align with our results.

We identified key features that are important across both our proposed model and XGBoost model. In the tabular data, demographics like BMI, weight, age, GPS information, and sleep quality consistently rank within the top 20 contributing features for both methods. However, when analyzing Fitbit channels, XGBoost places greater importance on heart rate and calorie-related metrics, while steps and distance are less influential. In contrast, deep learning models prioritize steps and distance over heart rate and calories.

The contributing features for stress detection in LifeSnaps and PMData vary between our model and XGBoost. In PMData, demographic factors such as age and gender play a major role in stress prediction for our model, whereas XGBoost assigns them less importance. In LifeSnaps, our model prioritizes step goal labels, BMI, and location, while XGBoost focuses more on heart rate, age, and personality traits. The contributing features differ because tree-based models like XGBoost prioritize high-information-gain splits, making them effective at capturing simple feature interactions, while deep learning models rely on gradient-based optimization and learn hierarchical representations that may emphasize different feature combinations.

4.4. Data Bias Analyses

In this section, we analyze the bias present in the LifeSnaps and PMData datasets. We focus on the bias in the stress labels, which are derived from self-reported surveys. The LifeSnaps dataset contains 228 labeled data points, while the PMData dataset has 2079 labeled data points. The sensitive attributes in both datasets are gender (female or male) and age group (under 30 or over 30).

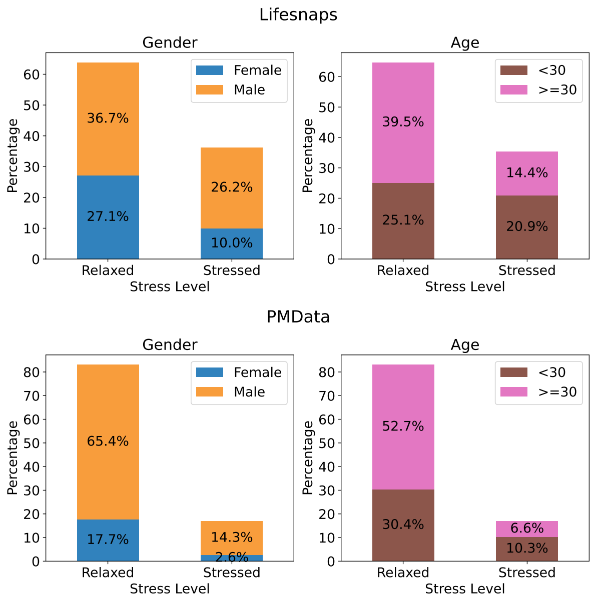

Figure 8: The proportion of gender and age groups across different target label values on the LifeSnaps and PMData datasets.

Figure 8 shows that male participants consistently report higher stress levels across both datasets. Additionally, younger participants (under 30) tend to report higher stress compared to older individuals.

Table 3: T-test and correlation results of gender and age (under 30 vs over 30) with stress in LifeSnaps and PMData datasets.

	LifeSnaps	
	t-test	corr. coef.
Gender	$t = -2.112, p < 0.05$	$t = 0.1399, p < 0.05$
Age	$t = -2.879, p < 0.01$	$t = -0.1953, p < 0.01$
	PMData	
	t-test	corr. coef.
Gender	$t = -2.757, p < 0.01$	$t = 0.0556, p < 0.05$
Age	$t = -8.178, p < 0.001$	$t = -0.1869, p < 0.001$

Table 3 presents the t-test and Pearson correlation results for gender and age relationships with stress levels across both datasets. Statistical analysis reveals significant differences in stress levels between males and females in both LifeSnaps and PMData datasets. Additionally, a significant negative correlation exists between age and stress in both datasets, with younger participants (under 30) consistently reporting higher stress levels compared to older participants.

4.5. Ablation Studies

In ablation studies, all results are derived under mixing user split with generalized models.

4.5.1. Effect of Single vs Multichannel Fitbit Data

Table 4: The values of test AUC in percentage using a single Fitbit channel and using all channels, in combination with tabular features. All values are derived using all the unlabeled data and labeled data, and random splitting generalized models. The results are from 20 experiments with 5-fold cross-validation and 4 random seeds, under the setting of random splitting with generalized models. Hyperparameter optimization is performed independently for each fold and each seed to obtain the best configuration.

	LifeSnaps	PMData
Steps	73.63 ± 3.38	81.35 ± 3.10
Heart Rate	67.84 ± 4.76	81.61 ± 2.47
Calories	69.77 ± 3.68	81.89 ± 2.93
Distance	73.75 ± 4.72	81.71 ± 3.10
Temperature	65.52 ± 5.31	-
All	72.45 ± 4.70	81.14 ± 1.79

To evaluate the effectiveness of using multiple Fitbit channels, we have conducted experiments with a single Fitbit channel and tabular features for self-supervised multimodal learning, using all unlabeled and labeled data in both datasets. The results are shown in Table 4. We found that focusing on key channels like steps and distance improves model performance compared to using all channels in the LifeSnaps dataset. On the other hand, using less important channels such as heart rate and calories resulted in worse performance than using all channels. We performed a Tukey's HSD test for pairwise group comparisons to assess performance across channels. For the LifeSnaps dataset, models trained using steps or distance showed significantly better test AUCs ($p < 0.05$) compared to models trained on other channels. In contrast, in the PMData dataset, the prediction performance was similar when using all channels or a single channel, with results consistently around 81.5 percent. This may be because, unlike the LifeSnaps dataset, the hourly features in the PMData dataset have less influence on stress, and the tabular features provide more comprehensive insights into stress levels.

4.5.2. Influence of Pretraining Data Size

We conduct ablation studies to evaluate the effectiveness of varying percentages of training data. The results are shown in Table 5. Please note that we used different approaches for LifeSnaps and PMData. In LifeSnaps, where most of the data is unlabeled, the pretraining proportion indicates the percentage of unlabeled data utilized for pretraining. On the other hand, PMData contains mostly labeled data, with 2811 labeled data points and only 327 unlabeled. For the PMData experiments, we fine-tuned the model using 30 percent of the labeled data and treated the remaining 70 percent as unlabeled after we pretrained the model on varying percentages of the unlabeled data.

However, for both the LifeSnaps and PMData datasets, varying the percentage of data used for pretraining did not show significant differences in test AUC. This might be due to the limited amount of data in both datasets.

5. Limitations and Future Work

Our study has several limitations. First, contrastive pretraining methods requires a large amount of unlabeled data to learn effective representations. While we demonstrate the effectiveness of our method on stress detection tasks with limited labeled samples, the enhancement of test AUC after contrastive pretraining is small on both experimental datasets. The modest performance gain us-

Table 5: The values of test AUC in percentages versus samples of data used for pretraining. L represents the number of labeled data points used for fine-tuning. U represents the number of unlabeled data points for pretraining. We show the number of labeled and unlabeled data points for LifeSnaps (LS) and PMData (PM) respectively for each data proportion setting. The numerical results are derived with 5-fold cross-validationn and 4 random seeds under the setting of non-random splitting with generalized models. Note that hyperparameter optimization was not performed for the ablation studies, which leads to slightly lower test AUC values compared to those reported in Table 1, where optimization was applied.

Proportion	LS (U / L)	PM (U / L)	LifeSnaps	PMData
1%	56 / 228	638 / 621	68.63 ± 4.81	65.39 ± 3.12
10%	562 / 228	798 / 621	68.40 ± 5.51	63.68 ± 2.54
50%	2811 / 228	1509 / 621	67.61 ± 5.45	64.20 ± 2.34
100%	5622 / 228	2397 / 621	67.62 ± 7.07	64.10 ± 2.07

ing self-supervised learning is expected as we do not have much training data. The model requires a large amount of data to achieve good performance, however, the data size in stress detection applications is usually small. To tackle the limited amount of data, we plan to explore how leveraging representations from general tasks in deep learning and large language models can enhance our model's performance on stress detection datasets. We can also consider training our model on a combination of multiple stress datasets to increase the overall data volume. Furthermore, we can collect large-scale data from multiple sources and scale up our model size and training time to improve the prediction performance.

Second, on the architecture of our proposed model, the structured text prompts generated from the data are based on statistical features and tabular features, which may not be the best prompt generation method. In future work, we will explore more sophisticated learning-based methods for generating prompts that capture the underlying patterns of the data more effectively. In addition, the multimodal encoder architecture could be further optimized to enhance the integration of time series and tabular features. Future research could investigate more advanced multimodal architectures that leverage the strengths of each modality more effectively.

Lastly, while our current study focuses on wearable data, we note that mobile phones can also collect physiological and behavioral signals, and we plan to extend our model to incorporate mobile phone data in future work to enhance its accessibility. Additionally, intervention strategies based on the model's predictions are crucial for real-world applications. While this is beyond the scope of our current work, we recognize its importance and aim to explore this direction in future research.

6. Conclusion

In this study, we proposed a contrastive pretraining method for stress detection using multimodal data, including wearable time series data and demographic and contextual data. Our approach leverages the complementary information from time series and tabular features to learn representations that capture the underlying patterns of stress. By generating structured text descriptions of the data and employing contrastive pretraining, we align multimodal data representations and learn discriminative representations for stress detection. We evaluated our method on two multimodal datasets: LifeSnaps and PMData, and demonstrated its effectiveness in stress detection tasks.

7. Acknowledgements

This project was funded by NIH (#R01DA059925) and NSF (#2047296).

References

Cristian-Paul Bara, Michalis Papakostas, and Rada Mihalcea. A deep learning approach towards multimodal stress detection. In *Affective Computing @ AAAI*, pages 67–81, 2020.

Carrie E Bartley and Scott C Roesch. Coping with daily stress: The role of conscientiousness. *Personality and individual differences*, 50(1):79–83, 2011.

Pramod Bobade and M Vani. Stress detection with machine learning and deep learning using multimodal physiological data. In *2020 Second International Conference on Inventive Research in Computing Applications (ICIRCA)*, pages 51–57. IEEE, 2020.

Leo Breiman. Random forests. *Machine learning*, 45: 5–32, 2001.

Yekta Said Can, Bert Arnrich, and Cem Ersoy. Stress detection in daily life scenarios using smart phones and wearable sensors: A survey. *Journal of biomedical informatics*, 92:103139, 2019.

Tianqi Chen and Carlos Guestrin. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, pages 785–794, 2016.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International Conference on Machine Learning*, pages 1597–1607. PMLR, 2020.

Sumit Chopra, Raia Hadsell, and Yann LeCun. Learning a similarity metric discriminatively, with application to face verification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, volume 1, pages 539–546. IEEE, 2005.

Orianna DeMasi, Konrad Kording, and Benjamin Recht. Meaningless comparisons lead to false optimism in medical machine learning. *PloS one*, 12(9):e0184604, 2017.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

Linus Ericsson, Henry Gouk, Chen Change Loy, and Timothy M Hospedales. Self-supervised representation learning: Introduction, advances, and challenges. *IEEE Signal Processing Magazine*, 39(3):42–62, 2022.

José Raúl Machado Fernández and Lesya Anishchenko. Mental stress detection using bioradar respiratory signals. *Biomedical signal processing and control*, 43: 244–249, 2018.

Prerna Garg, Jayasankar Santhosh, Andreas Dengel, and Shoya Ishimaru. Stress detection by machine learning and wearable sensors. In *Companion Proceedings of the 26th International Conference on Intelligent User Interfaces*, pages 43–45, 2021.

Roger Garriga, Javier Mas, Semhar Abraha, Jon Nolan, Oliver Harrison, George Tadros, and Aleksandar Matic. Machine learning model to predict mental health crises from electronic health records. *Nature medicine*, 28(6): 1240–1248, 2022.

Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Guo, Mohammad Gheshlaghi Azar, et al. Bootstrap your own latent-a new approach to self-supervised learning. *Advances in Neural Information Processing Systems*, 33:21271–21284, 2020.

Richard A Groeneveld and Glen Meeden. Measuring skewness and kurtosis. *Journal of the Royal Statistical Society Series D: The Statistician*, 33(4):391–399, 1984.

Megha V Gupta, Shubhangi Vaikole, Ankit D Oza, Amisha Patel, Diana Petronela Burduhos-Nergis, and Dumitru Doru Burduhos-Nergis. Audio-visual stress classification using cascaded rnn-lstm networks. *Bioengineering*, 9(10):510, 2022.

Tanvir Islam and Peter Washington. Individualized stress mobile sensing using self-supervised pre-training. *Applied Sciences*, 13(21):12035, 2023.

Sneha Jha, Erik Mayer, and Mauricio Barahona. Improving information fusion on multimodal clinical data in classification settings. In *Proceedings of the 13th International Workshop on Health Text Mining and Information Analysis (LOUHI)*, pages 154–159, 2022.

Eui-Joong Kim and Joel E Dimsdale. The effect of psychosocial stress on sleep: a review of polysomnographic evidence. *Behavioral sleep medicine*, 5(4): 256–278, 2007.

Hye-Geum Kim, Eun-Jin Cheon, Dai-Seg Bai, Young Hwan Lee, and Bon-Hoon Koo. Stress and heart rate variability: a meta-analysis and review of the literature. *Psychiatry investigation*, 15(3):235, 2018.

Narine Kokhlikyan, Vivek Miglani, Miguel Martin, Edward Wang, Bilal Alsallakh, Jonathan Reynolds, Alexander Melnikov, Natalia Kliushkina, Carlos Araya, Siqi Yan, et al. Captum: A unified and generic model interpretability library for pytorch. *arXiv preprint arXiv:2009.07896*, 2020.

Russell Li and Zhandong Liu. Stress detection using deep neural networks. *BMC Medical Informatics and Decision Making*, 20:1–10, 2020.

Scott M Lundberg and Su-In Lee. A unified approach to interpreting model predictions. *Advances in Neural Information Processing Systems*, 30, 2017.

Duncan McElfresh, Sujay Khandagale, Jonathan Valverde, Vishak Prasad C, Ganesh Ramakrishnan, Micah Goldblum, and Colin White. When do neural nets outperform boosted trees on tabular data? *Advances in Neural Information Processing Systems*, 36, 2024.

Suha Rabbani and Naimul Khan. Contrastive self-supervised learning for stress detection from ecg data. *Bioengineering*, 9(8):374, 2022.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, pages 8748–8763. PMLR, 2021.

Aniruddh Raghu, Payal Chandak, Ridwan Alam, John Guttag, and Collin Stultz. Sequential multi-dimensional self-supervised learning for clinical time series. In *International Conference on Machine Learning*, pages 28531–28548. PMLR, 2023.

Pritam Sarkar and Ali Etemad. Self-supervised ecg representation learning for emotion recognition. *IEEE Transactions on Affective Computing*, 13(3):1541–1554, 2020.

Neil Schneiderman, Gail Ironson, and Scott D Siegel. Stress and health: psychological, behavioral, and biological determinants. *Annu. Rev. Clin. Psychol.*, 1(1): 607–628, 2005.

Wonju Seo, Namho Kim, Cheolsoo Park, and Sung-Min Park. Deep learning approach for detecting work-related stress using multimodal signals. *IEEE Sensors Journal*, 22(12):11892–11902, 2022.

Aditi Sharma, Kapil Sharma, and Akshi Kumar. Real-time emotional health detection using fine-tuned transfer networks with multimodal fusion. *Neural computing and applications*, 35(31):22935–22948, 2023.

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

Vajira Thambawita, Steven Alexander Hicks, Hanna Borgli, Håkon Kvale Stensland, Debesh Jha, Martin Kristoffer Svensen, Svein-Arne Pettersen, Dag Johansen, Håvard Dagenborg Johansen, Susann Dahl Pettersen, et al. Pmdata: a sports logging dataset. In *Proceedings of the 11th ACM Multimedia Systems Conference*, pages 231–236, 2020.

Yujin Wu, Mohamed Daoudi, and Ali Amad. Transformer-based self-supervised multimodal representation learning for wearable emotion recognition. *IEEE Transactions on Affective Computing*, 15(1): 157–172, 2023.

Sofia Yfantidou, Christina Karagianni, Stefanos Efstathiou, Athena Vakali, Joao Palotti, Dimitrios Panteleimon Giakatos, Thomas Marchioro, Andrei Kazlouski, Elena Ferrari, and Šarūnas Girdzijauskas. Lifesnaps, a 4-month multi-modal dataset capturing unobtrusive snapshots of our lives in the wild. *Scientific Data*, 9(1): 663, 2022.

Han Yu and Akane Sano. Semi-supervised learning for wearable-based momentary stress detection in the wild. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 7(2):1–23, 2023.

Han Yu, Huiyuan Yang, and Akane Sano. Leaves: learning views for time-series data in contrastive learning. *arXiv preprint arXiv:2210.07340*, 2022.

Han Yu, Peikun Guo, and Akane Sano. Ecg semantic integrator (esi): A foundation ecg model pretrained with llm-enhanced cardiological text. *Transactions on Machine Learning Research*, 2024.