Electrodermal Sensing-Based Non-Invasive Context-Aware Dehydration Alert System Using Machine Learning Algorithm

K T Tharun, P Raahul Prasath, P Mounaguru, S Kathirvarshan and R Sruthika

September 6, 2022

Electrodermal sensing-based Non-Invasive Context-Aware Dehydration Alert System Using Machine Learning Algorithm

Tharun K T, Raahul Prasath P, Mounaguru P, Kathirvarsan S, Sruthika R

Department of ECE,Bannari Amman Institute of Technology, Sathyamangalam.

Abstract:

Staying hydrated and drinking fluids is extremely crucial to stay healthy and maintaining even basic bodily functions. Studies have shown that dehydration leads to loss of productivity, cognitive impairment and mood in both men and women. However, there are no such existing tools that can monitor dehydration continuously and provide alert to users before it effects on their health. In this paper, we propose to utilize wearable Electrodermal Activity (EDA) sensors in conjunction with signal processing and machine learning techniques to develop first time ever a dehydration self-monitoring tool, Monitoring My Dehydration (MMD), that can instantly detect the hydration level of human skin. Moreover, we develop an Android application over

Bluetooth to connect with wearable EDA sensor integrated wristband to track hydration levels of the user's real time and instantly alert to the users when the hydration level goes beyond the danger level. To validate our developed tool's performance, we recruit 5 users, carefully designed the water intake routines to annotate the dehydration ground truth and trained state-of-art machine learning models to predict instant hydration level i.e., well-hydrated, hydrated, dehydrated and very dehydrated.

I. Introduction:

The Present innovation generally deals with A Non-Invasive Dehydration Monitoring and Alert System Using Electro dermal Activity. Hydration level is a strong indicator of health that can help improve medical implications on potential health hazards and it is extremely important to track the hydration level (HL) of human body, specifically for children, the elderly and patients with underlying medical conditions such as diabetes. Despite increased risks of disability, mortality and hospital admissions associated with water-loss, dehydration is often unnoticed due to lack of immediate symptoms and instant measurement that necessitates the dehydration measurement tool significantly.

This work combines the EDA sensor of the E4 device with common machine-learning models and basic Android application frameworks to build a comprehensive application to detect user hydration levels and alert the user that he should be drinking water. The data collection methods are described, along with difficulties encountered with both the lack of available participants and technical issues from wearing the device for long periods of time. Once this raw data is acquired, it has to be pre-processed; the pre-processing is described, and then details are given on all the different techniques used to train models.

II. Methodologies :

A. Detecting Dehydration

Dehydration detection using mobile sensors is a relatively new concept. Users in good health were recruited to initially perform a cognitive task known as the Stroop Task while being fully hydrated. During the course of the task, EDA and Pressure Relief Value readings were collected using a wearable sensor. The participants were then instructed to not consume liquids or water-heavy foods for the next 24 hours. Upon their return, they were instructed to perform the same task while wearing the sensors. Finally, the participants rehydrated and performed the same task once more. The authors then used various machine learning methods, such as logistic regression, support vector machines, decision trees, and K-Nearest Neighbor (KNN) classifiers to model the data and predict the dehydration of the user. Additionally, a variety of physiological measures were taken of the subjects to confirm that they were, indeed, mildly dehydrated. The authors mention at the end of the paper that one of the shortcomings of their study is that the hydration was only being sensed in a very controlled environment, and that it would be valuable future work to assess the accuracy of similar methods at determining hydration levels in less controlled environments. Another somewhat-similar paper is concerned with extracting features from EDA data using a variety of methods and developed a new algorithm for the fast and efficient interpretation of EDA data into EDA.

B. Overall Framework

The flow starts with the Empatica E4 device, which collects raw data about the user and doessome basic preprocessing (for example, calculating Skin Conductance from EDA signals,non-negative sparse deconvolution to extract components of EDA signal). It then sends this data to the user's smartphone over Bluetooth. The Android application on the user's phone then preprocesses the data by using basic statistical methods to help remove noise from the data. Once this is done, the data is fed to a pre-trained machine learning model using the Waikato Environment for Knowledge Analysis (WEKA) Java library. Then, if the machine learning model predicts a change in hydration level, it will trigger a method that sends the user anotification to alert them of their changed hydration level.

C. Electrodermal Activity Feature Extraction

Electrodermal activity also known as skin conductance measurement over time includes two components. (i) Skin conductance Base Level (SBL), which changes slowly over time (tonic changes) and indicates the general activation of the sympathetic nervous system, (ii) Skin Conductance Responses (SCRs), changes that last for shorter periods (phasic changes). SCRs indicate the activation of the somatic nervous system (SNS) but also reflect responses to events that are new, unexpected, relevant, and/or aversive. Using EDA data to measure arousal in a continuous stimulus setting requires three steps in data processing and analysis. First step is pre-processing which involves data cleaning, filtering, down sampling, cutting, smoothing, artifact correction and decomposition of the signal into its tonic and phasic components. The SBL is typically approximated by frequency filtering, statistical modeling or simple linear interpolation between the skin conductance measures that are not overlaid by responses. The second step is parameterization, which involves deciding which parameter of the EDA data to measure/calculate. For a phasic parameter, this process includes massive abstraction of the phasic signal component, for example, counting responses. The third step is the correlation of the extracted data with the stimulus. We used LedaLab toolbox for EDA data preprocessing and extracting features. We employed butterworth low-pass filter, hanning smoothing with window size 4 and manual movement artifact correction. We decomposed EDA data into its tonic and phasic components using Continuous

Decomposition Analysis (CDA) and Discrete Decomposition Analysis (DDA) Continuous Decomposition Analysis (CDA): This method helps extract the phasic (driver) information underlying the EDA signal, and aims at retrieving the signal characteristics of then underlying sudomotor nerve activity (SNA). EDA data is deconvolved by the general response shape which results in a large increase of temporal precision and then data is being decomposed into continuous phasic and tonic components. This helps compute the several standard features of phasic EDA. We tracked the related events as our pre-labeled activities and extracted 7 time-domain features from CDA. We used standard deviation, mean and variances on these features over the activity window.Discrete Decomposition Analysis (DDA): This method decomposes EDA data into distinct phasic components and a tonic component by means of Nonnegative Deconvolution. The methodhelps capture and explore all intra-individual deviations of the general response shape and compute a detailed full model of all components in the entire data set. This method is particularly suited for physiological models of the SCR. We extracted 5 features from DDA for each activity window and extracted statistical mean, variance and standard deviation on these over the activity session.After data pre-processing and feature extraction, various different machine learning and deep learning algorithms are applied, including the hybrid approach to estimate the skin hydration.

The data was initially collected for the intervals of 5 to 30 minutes and then spitted into smallest segments using a window operation. The window size, W, represents the size of each segment in seconds. Subsequently, feature extraction is performed on the segmented data. It is worth mentioning that different window sizes produce different data pattern after feature extraction. Considering this, an important task would be to identify the optimal window size that produces best results. A feature space, F, of following nine statistical features is used: F∈ {Minimum, Mean, Standard Deviation, Percentile, Median, Kurtosi}. The values of each of the aforementioned features are calculated for the window sizes of 30 and 60 seconds. After feature extraction, it is important to determine the combination of features which generates the best performance for estimation of skin hydration. For that purpose, a genetic algorithms is applied to evaluate all combinations of the features for each algorithm. The data is segmented for each window size and above-mentioned nine features are extracted from each segment. For instance, when a window size of 30 seconds is selected, the data is segmented into non-overlapping segments of 30 seconds and features are extracted from each segment of 30 seconds data. This creates a vector of nice feature for each segment. Using these feature, the data-set is created.

D. User Interface and Notifications

The ‘Hydration Alert’ application is relatively simple and is composed of the ‘Main Activity’, a custom defined Service, and various other custom classes and enumerations. The main activity has three important functions. First, it is responsible for binding the Empatica E4 device service to the Android application, and handling any connection issues. Second, the main activity manages the UI for the device, displaying the relevant information in a clear, easy-to-understand interface. Third, it has the functions for creating and sending notifications; whenever the user’s predicted hydration level changes, a callback function is triggered that both modifies the UI and also sends the user a notification. Whether the application may be active in the foreground or the background, connection always needs to be enabled until the application has been terminated. Once connected, the phone initially calculates the hydration level and then displays it along with a visual cue of a high or low water level. The Empatica framework used for connecting to and receiving data from the device is implemented as a custom Android Service, implementing their public application programming interfaces. This interface allows the Android service to run in the background, even after the application is suspended for continuous monitoring of the user’s EDA levels. A custom interface is defined within the Service as well to allow for the creation of delegate callback methods used by the main activity.

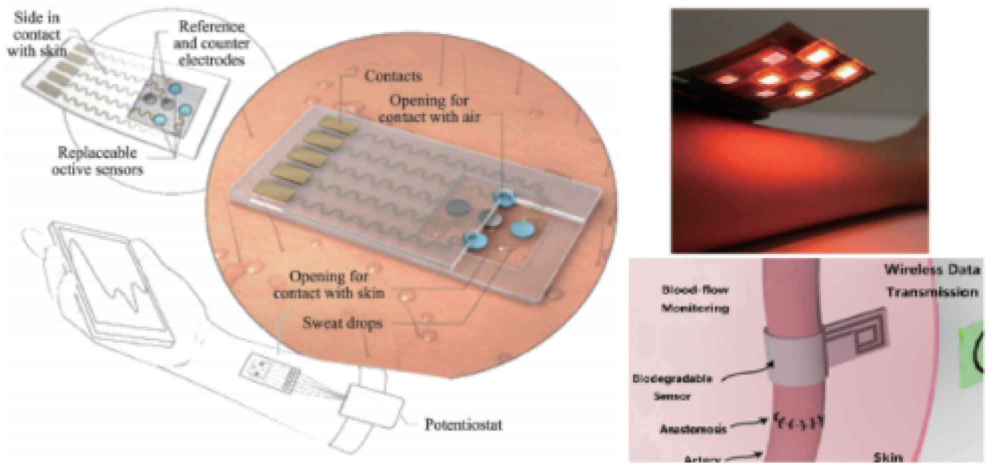

Figure 1 Detailed block diagram of the wearable blood sensor

The Blood sensors, alternating arrays of printed light-emitting diodes and photodetectors, can detect blood oxygen levels in any part of the body. The sensor uses light-emitting diodes to emit red and near infrared light, penetrating the skin and detecting the proportion of reflected light. The sensor made of biodegradable materials utilizes edge-field capacitance technology to monitor arterial blood and then transmits the data wirelessly.

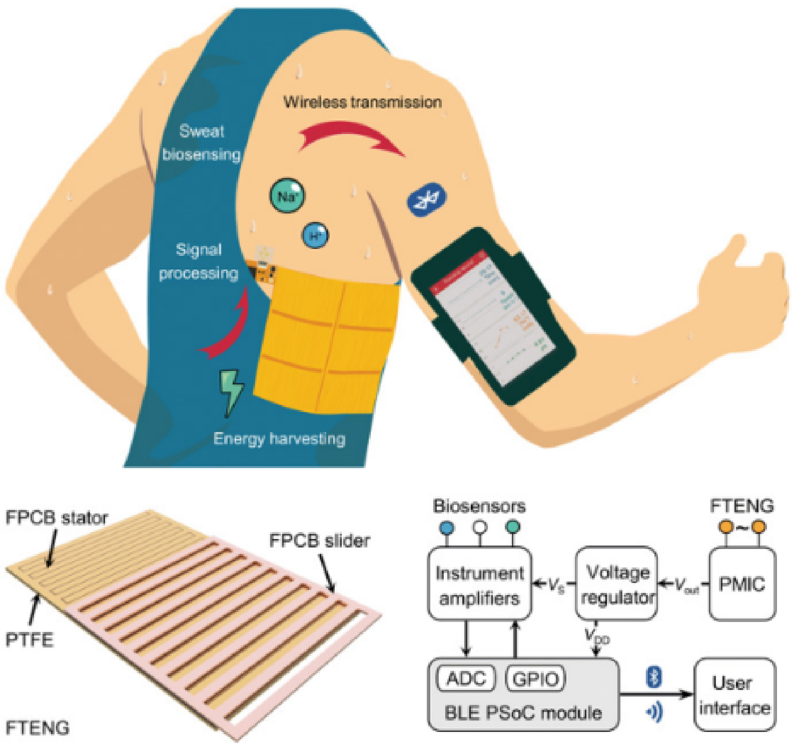

Figure 2 Detailed block diagram of the wearable sweat sensor

The Schematic illustrating the FWS that integrates human motion energy harvesting, signal processing, microfluidic-based sweat bio sensing, and Bluetooth-based wireless data transmission to a mobile user interface for real-time health status tracking and the System-level block diagram showing the power management, signal transduction, processing, and wireless transmission of the FWS3 from the FTENG to the biosensors, then to the user interface.

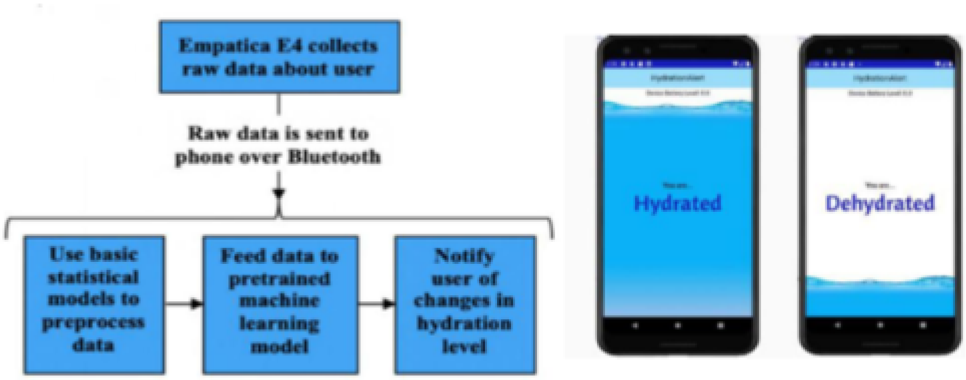

Figure 3 MMD Hydration Alert Android System Overview and Application's user interface

The MMD Hydration Alert Android System Overview. The overall project layout is shown in Fig. 3. The flow in Fig. 3 starts with the Empatica E4 device, which collects raw data about the user and does some basic preprocessing. It then sends this data to the user's smartphone over Bluetooth. The Android application on the user's phone then preprocesses the data by using basic statistical methods to help remove noise from the data. We can see the interface the user will see when the user connects the device to the phone. Once connected, the phone initially calculates the hydration level and then displays it along with a visual cue of a high or low water level. The Empatica framework used for connecting to and receiving data from the device is implemented as a custom Android Service, implementing their public application programming interfaces.

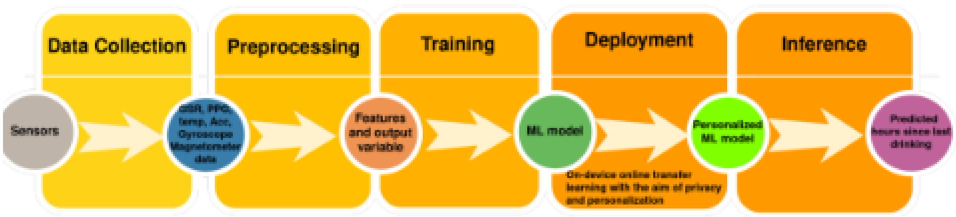

Figure 4 Schematic diagram of Simple Adaptive Multi-label Activity Recognition Framework

The Adaptive Multi-label activity recognition which can provide instant postural and drinking water events classification. However, our ultimate goal is to detect instant hydration level using EDA signal. We first extract 12 EDA features from EDA raw signal and fed them into machine learning models.

III.Conclusion:

In this study we have proposed a Electrodermal sensing-based Non-Invasive Context-Aware Dehydration Alert System Using Machine Learning Algorithm Detection.Through this we can validate our developed tool's performance.

IV. References:

1. "Hydration Assessment of Athletes," Oct. 2006. Accessed on: May 11, 2020. [Online].

Available: https://www.gssiweb.org/sports-science-exchange/article/sse-97-hydration-assessment-of-athletes.

2. Mendelson, Y.; Dao, D.K.; Chon, K.H. Multi-channel pulse oximetry for wearable physiological monitoring. In Proceedings of the 2013 IEEE International Conference on Body Sensor Networks, Cambridge, MA, USA, 6-9 May 2013; pp. 1--6.

3. H. F. Posada-Quintero, N. Reljin, A. Moutran, D. Georgopalis, E. C.-H. Lee, G. E. W. Giersch, D. J. Casa, and K. H. Chon, "Mild Dehydration Identification Using Machine Learning to Assess Autonomic Responses to Cognitive Stress," Nutrients, vol. 12, no. 1, p. 42, 2019.

4. Yoneda, K. Anatomic pathway of fluid leakage in fluid-overload pulmonary edema in mice. The Am. journal pathology 101, 7 (1980).

5. Veiga, D. et al. Postoperative delirium in intensive care patients: risk factors and outcome. Braz. J. Anesthesiol. 62, 469–483 (2012).

6. Prowle, J. R., Echeverri, J. E., Ligabo, E. V., Ronco, C. & Bellomo, R. Fluid balance and acute kidney injury. Nat. Rev. Nephrol. 6, 107–115 (2010).

7. Wizemann, V. et al. The mortality risk of overhydration in haemodialysis patients. Nephrol. Dial. Transplantation 24, 1574–1579 (2009).

8. Boyd, J. H., Forbes, J., Nakada, T.-a., Walley, K. R. & Russell, J. A. Fluid resuscitation in septic shock: a positive fluid balance and elevated central venous pressure are associated with increased mortality. Critical care medicine 39, 259–265 (2011).

9. Fortes, M. B. et al. Is this elderly patient dehydrated? diagnostic accuracy of hydration assessment using physical signs, urine, and saliva markers. J. Am. Med. Dir. Assoc. 16, 221–228 (2015).

10. Armstrong, L. E. Hydration assessment techniques. Nutr. reviews 63, S40–S54 (2005).