A Machine Learning Approach to Detecting Low Medication State with Wearable Technologies

Andy Cheon[1], Stephanie Yeoju Jung[1], Collin Prather[1], Matthew Sarmiento[1], Kevin Wong[1] and Diane Myung-kyung Woodbridge[2]

***Abstract*— Medication adherence is a critical component and implicit assumption of the patient life cycle that is often violated, incurring financial and medical costs to both patients and the medical system at large. As obstacles to medication adherence are complex and varied, approaches to overcome them must themselves be multifaceted.**

This paper demonstrates one such approach using sensor data recorded by an Apple Watch to detect low counts of pill medication in standard prescription bottles. We use distributed computing on a cloud-based platform to efficiently process large volumes of high-frequency data and train a Gradient Boosted Tree machine learning model. Our final model yielded average cross-validated accuracy and F1 scores of 80.27% and 80.22%, respectively.

We conclude this paper with two use cases in which wearable devices such as the Apple Watch can contribute to efforts to improve patient medication adherence.

***Index Terms*— Medication Adherence, Machine Learning, Distributed Computing, Wearable Sensors**

I. Introduction

In 2003 the World Health Organization estimated that among developed countries, approximately 50% of patients with chronic illnesses do not take their medication as prescribed [1]. The issue of medication adherence, while readily acknowledged in medical circles, continues to persist as rates of chronic illness increase worldwide [2]. Further compounding the issue, non-compliance with medication schedules puts a significant strain on healthcare systems. In the United States alone, hospitalizations due to medication nonadherence are estimated to cost healthcare systems $289 billion annually [3]. A system that provides accurate estimates of whether or not a patient has taken their medication has the potential to drastically improve patient outcomes, saving the healthcare industry billions of dollars in the process.

An increasing number of technologies have emerged in recent years to promote strict adherence to prescribed medication schedules, many of which have been shown to mitigate nonadherence and improve patient outcomes [4][5]. Some of the earliest approaches involved mobile applications that remind users to consume medication at the prescribed time. More recently, applications have added support for smart wearable devices. As these devices have grown smaller, lighter, and less intrusive to the lives of wearers, increasing efforts have been made to leverage the newly available data to promote medication adherence [2][3]. The majority of efforts have focused on developing reliable means of verifying the consumption of patient medication. Success in this area gives reason to consider how we may further develop complementary technologies to address the issue of medication adherence from all angles.

With this in mind, we focus our attention on one neglected approach to resolving the problem of medication nonadherence: ensuring the delivery of refillment medication before a patient's supply of medication has been completely consumed. We are motivated in particular by patients dependent on daily medication, for whom a single missed refillment poses genuine peril.

In this paper we outline a machine learning-based approach to detecting low counts of pill medication in standard prescription bottles using sensor data from an Apple Watch. Using data from the watch's gyroscope and accelerometer, together with audio decibel levels, we develop a classifier that accurately predicts when there is a low number of pills remaining at the time a pill is retrieved from a prescription bottle. Our study includes 16 subjects, each performing 5 trials of retrieving a pill from a prescription bottle at 8 different levels of pill medication in the bottle. We excluded 21 trials due to technical difficulties during the experiment, leaving us with 619 total trials. This study was carried out with approval by the Institutional Review Board for the Protection of Human Subjects at the University of San Francisco.

II. Related Work

Existing recent research on medication adherence using wearables focuses primarily on detecting the motions around pill intake. Chen et al. [6] developed a medication adherence monitoring system for pill intake from pharmacy bottles using kinetic cameras and inertial sensors that focused on detecting "twist-cap" and "hand-to-mouth" actions.

Kalatarian et al. [7] focused on detecting the distinct actions of "opening a pill bottle" and "pouring pills into hand" also using both gyroscope and accelerometer data collected from a smartwatch at 16 Hz. Predictions were made for both actions, and if "pouring pills into hand" occurred within six seconds after "opening a pill bottle", then a subject was classified as having successfully taken a pill. Upon comparing their model's predictions of pill-taking with similar activities such as raising one's arm or opening

[1] Andy Cheon, Stephanie Yeoju Jung, Collin Prather, Matthew Sarmiento and Kevin Wong are with the MS in Data Science Program, University of San Francisco. These authors contributed equally. {acheon, yjung7, cjprather, msarmiento3, kbwong2}@usfca.edu

[2] Diane Myung-kyung Woodbridge, Ph.D. is an assistant professor at the MS in Data Science Program, University of San Francisco. dwoodbridge@usfca.edu

a water bottle, they found that the gyroscope is particularly effective in distinguishing the motions of taking a pill from other similar activities.

Fozoonmayeh et al. [8][9] developed a smartwatch based medication intake detection system also using gyroscope and accelerometer sensors that was able to classify pill-taking among other activities such as writing, walking, and drinking water with an F1 score of 98%. This explicitly demonstrated the feasibility of classifying when a subject takes a pill using a smartwatch; however this work did not detect the number of pills remaining, as no labels were present.

III. Data Collection

Experimental data was collected from a total of 16 subjects using the watchOS application *SensorLog* [10] on an Apple Watch. The application recorded 79 distinct variables at a frequency of 100 Hz, including gyroscope and accelerometer data. Each experimental trial was conducted as follows. Using pill substitutes in a standard pharmaceutical prescription container, subjects picked up and opened the container, extracted and consumed a single pill, then closed the container. Each subject, monitored by two experimenters, conducted the experiment sitting at a table in a room isolated from external noise, allowing for accurate recording of audio decibel levels. Each subject performed five trials of the experiment at eight different levels of pill counts: 0, 1, 5, 10, 15, 20, 25, and 30 pills. In the case of 0 pills, subjects simulated the action of pill consumption without consuming a pill substitute. Data were collected for this case to aid in isolating the feature importance of audio decibel levels from the remaining primarily motion-based features. Indeed, our final analysis demonstrates that gyroscopic data, paired with audio decibel levels, contain adequate signal for detecting low pill counts.

Each trial ranged from 8 to 12 seconds in length, producing 800 to 1200 readings for all 79 variables. Data were streamed to an iPhone and subsequently uploaded to a private Amazon Web Service (AWS) Simple Storage Service (S3) bucket [11].

Most important of all, a column of labels was added to each file indicating the number of pills present in the prescription container at the time a trial was performed. The label "1" was used to indicate a low medication state, which we define as the presence of ten pills or fewer. The label "0" was used to indicate an amount greater than 10 pills. A pill count threshold of 10 was chosen as it represents approximately one-third of the capacity of a standard prescription container.

IV. Data Preprocessing

The primary goal of preprocessing the collected data was to produce a single table for training from the 619 recorded CSV files. Recall that each trial recorded by *SensorLog* produced a single CSV file composed of 79 columns and 800 to 1200 rows. As each CSV file in its entirety represented a single instance of pill consumption, each needed to be transformed into a single row, i.e. training example, on which a model could be trained. A single row from any given trial prior to preprocessing did not in and of itself represent any identifiable action that a model could learn to identify, as it represents sensor readings at a single point in time.

We processed each file into a single row, as follows:

1) Partition the rows of each trial into 5 contiguous sections, each section further referred to as a time window. Each time window is almost exactly one-fifth the length of the original file.
2) Compute the summary statistics *mean*, *standard deviation*, *minimum*, and *maximum*, per feature, per time window. Each time window-feature combination maps to one new feature in the final table. For example, the minimum of gyroscopic data along the x-axis in time window 3 would produce a new feature: *min_gyro_x_3*.
3) Concatenate the resulting summary statistics into a new separate, single row that effectively summarizes the statistical properties of the entire trial – and is amenable to machine learning.

Our exploratory analysis led us to select 5 out of the 79 raw features collected – $gyro_x$, $gyro_y$, $gyro_z$, *audio peak power*, *audio average power* – in order to balance processing speed and predictive power, and the 4 summary statistics mentioned above – mean, standard deviation, minimum, and maximum – for a total of 5 features · 4 statistics · 5 windows + 2 metadata = 102 engineered features, resulting in a final table with 619 rows and 102 columns (Figure 1).

V. Distributed Machine Learning

Due to the potential need to scale massive amounts of incoming data, processing and modeling capabilities were built using an Apache Spark distributed computing platform [12] hosted on Amazon's Elastic Map Reduce (EMR) service [13]. Apache Spark allows all preprocessing functions to transform raw watch data into a format suitable for training a model in a matter of seconds. Additionally, it grants access to Spark's built-in machine learning platform, MLlib, providing access to state-of-the-art algorithms along with workflow utilities and model evaluation tools.

Each model trained using MLlib predicts the probability that a new, unseen trial corresponds to a low pill count. Binary predictions were derived from those probabilities by classifying any probability greater than or equal to 0.50 as *low* (10 or fewer), while any probability lower than 0.50 was classified as *high* (greater than 10, up to 30). To objectively evaluate the model quality, we considered classification accuracy and F1 score. Accuracy is defined as the proportion of correct predictions made out of total predictions. F1 score is defined as the harmonic mean of each model's precision and recall, which serves as a better measure of performance on unbalanced classes – which is marginally true in our case. F1 score is typically considered a better evaluation metric than accuracy as it strikes a balance between both false positive and false negative rate.

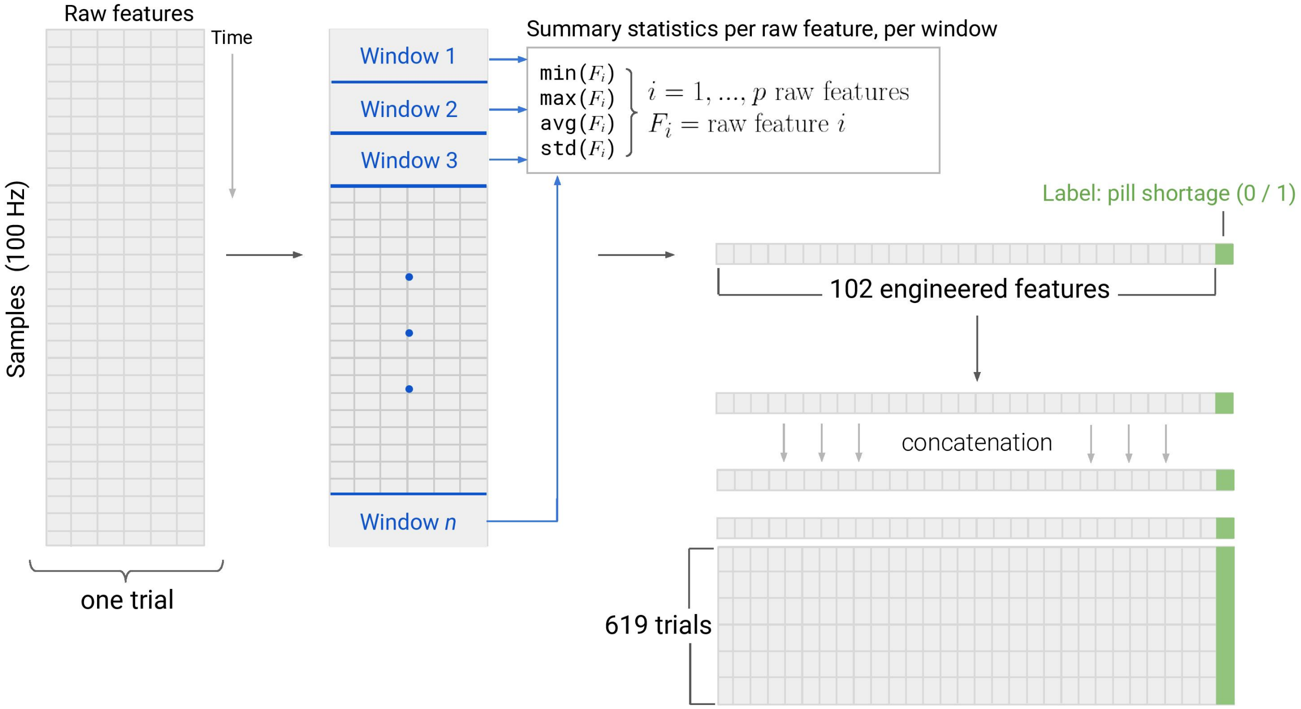

Fig. 1: Feature engineering process. Each of the 619 trials produced a data frame with columns consisting of 79 raw features recorded by SensorLog, and rows corresponding to values sampled at 100 Hz. The data frame was then divided into windows; for each window, four summary statistics were computed for each selected raw feature — $gyro_x$, $gyro_y$, $gyro_z$, *audio peak power*, *audio power* — resulting in a row vector of 102 engineered features. The 619 vectors were concatenated to form the training dataset for machine learning. Note that the data are unnormalized.

$$Precision = \frac{TP}{TP + FP} \tag{1}$$

$$Recall = \frac{TP}{TP + FN} \tag{2}$$

$$F1 = \frac{2 * (Recall * Precision)}{(Recall + Precision)} \tag{3}$$

A standard machine learning technique we employed to evaluate a model's ability to generalize to unseen data is *k-fold cross validation*, which splits the dataset into k equal-length folds, and trains k models on all but one fold each time. Evaluation metrics are computed k times for each model's performance on each of the holdout folds, and then averaged for a better approximation of the true performance on unseen data. We performed 5-fold cross validation resulting in folds of approximately 124 rows.

VI. Results

Our baseline logistic regression model [14] results in a classification accuracy of 69.92% and an F1 score of 69.20%. The Gradient-Boosted Tree model [15], comprised of 200 estimators using a tree-depth of 3 and learning rate of 0.30, reaches an improved accuracy of 80.27% and F1 score of 80.22%.

As shown in Figure 2, the models trained on data separated into five windows markedly outperform those trained on data separated into ten windows. The stages of motion each subject underwent while simulating medication intake fall into different windows from subject to subject due to natural variance in speed. Using fewer windows, subjects were more likely to be opening, consuming, or closing the prescription container within a single window – information which a tree-based model can better learn from and fit across all subjects.

VII. Conclusion

Contrary to prior belief, our results suggest that it is in fact possible to detect when a patient's pill-based medication is running low using only a smartwatch. With cross-validated accuracy and F1 scores of 80.27% and 80.22%, respectively, the Gradient-Boosted Tree model outperforms expectations.

It is important to note that this data was collected in a controlled setting, with minimal background noise. Subjects performed instructed movements, and in reality, patients take medication with far more variation in speed, movement, and background noise. A first solution to this problem would be to collect more data from a wide variety of subjects and environments, without any instructions on how to take their medication.

A separate avenue for improvement is to include more raw features, such as accelerometer sensor readings, in our

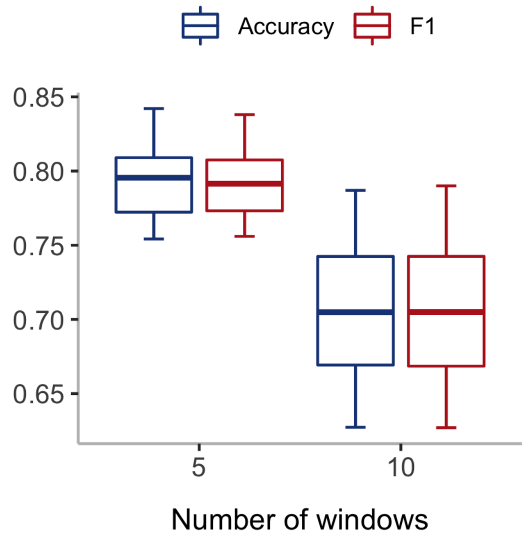

Fig. 2: Model evaluation. The Gradient-Boosted Tree model performs best when there are five windows, with an accuracy of 79.5%. Increasing the number of windows to 10 decreases median accuracy and F1 scores by approximately 11%.

preprocessing pipeline. *SensorLog* records 79 raw features, while our model makes use of five. This reduction of inputs was required due to the limited number of subject trial records and the nature of machine learning models – which suffer performance breakdowns when the number of columns within the training data set is larger than the number of rows. A consequence of transforming each record from long to wide was the creation of many more processed features, which forced us to limit the number of raw inputs. A sliding window approach, as opposed to the fixed windows we employ in this paper, may be capable of supporting larger quantities of features.

A third avenue for improvement is to spend more computational resources to fine-tune model hyperparameters. Currently, the Gradient-Boosted Tree model combined 200 estimators using a tree-depth of 3 and learning rate of 0.30. Grid search can be employed over a wider search space to further optimize the model. Moreover, it is possible that an entirely different model architecture, such as a recurrent neural network, could produce even better results; however, this would require an entirely separate preprocessing pipeline and training procedure.

Despite these considerations, our current machine learning model demonstrates the ability to predict a low medication state in a controlled environment with a high degree of accuracy, predicting correct state in four out of five cases. The external use cases of such work are numerous. One interesting use case could be to flag and alert medical systems of possible overdose events.

Another case could be an application for a smartwatch (such as the Apple Watch) that launches when triggered by a pre-programmed Near-Field-Communication (NFC) sensor attached to a prescription container. The application would then run two models simultaneously, a model to classify the subsequent actions as positive pill consumption, and a model to classify low medication state in the container – conditioned on prior positive pill consumption. When low medication state is detected, a medical system, doctor, or pharmacy would be notified. This would benefit proper tracking of medical adherence, while also allowing an automatic reordering system to be built into the application.

In summary, sensor data recorded by an Apple Watch were used to detect low counts of pill medication in standard prescription bottles with a machine learning model. Converting recorded data into a format amenable to machine learning involved splitting the data into a number of time windows, computing summary statistics within each window, and merging engineered features into a training dataset. Our results demonstrate the promise of wearable devices for developing new strategies to combat medical nonadherence.

REFERENCES

[1] M. T. Brown and J. K. Bussell, "Medication adherence: Who cares?" in *Mayo clinic proceedings*, vol. 86, no. 4. Elsevier, 2011, pp. 304–314.

[2] W. H. Organization *et al.*, *Adherence to long-term therapies: evidence for action*. World Health Organization, 2003.

[3] X. Liu, J. J. Lewis, H. Zhang, W. Lu, S. Zhang, G. Zheng, L. Bai, J. Li, X. Li, H. Chen *et al.*, "Effectiveness of electronic reminders to improve medication adherence in tuberculosis patients: a cluster-randomised trial," *PLoS medicine*, vol. 12, no. 9, 2015.

[4] M. Vervloet, A. J. Linn, J. C. van Weert, D. H. De Bakker, M. L. Bouvy, and L. Van Dijk, "The effectiveness of interventions using electronic reminders to improve adherence to chronic medication: a systematic review of the literature," *Journal of the American Medical Informatics Association*, vol. 19, no. 5, pp. 696–704, 2012.

[5] H. Kalantarian, B. Motamed, N. Alshurafa, and M. Sarrafzadeh, "A wearable sensor system for medication adherence prediction," *Artificial intelligence in medicine*, vol. 69, pp. 43–52, 2016.

[6] C. Chen, N. Kehtarnavaz, and R. Jafari, "A medication adherence monitoring system for pill bottles based on a wearable inertial sensor," in *2014 36th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*. IEEE, 2014, pp. 4983–4986.

[7] H. Kalantarian, N. Alshurafa, E. Nemati, T. Le, and M. Sarrafzadeh, "A smartwatch-based medication adherence system," in *2015 IEEE 12th International Conference on Wearable and Implantable Body Sensor Networks (BSN)*. IEEE, 2015, pp. 1–6.

[8] D. Fozoonmayeh, H. V. Le, E. Wittfoth, C. Geng, N. Ha, J. Wang, M. Vasilenko, Y. Ahn, and D. M.-k. Woodbridge, "A scalable smartwatch-based medication intake detection system using distributed machine learning," *Journal of Medical Systems*, vol. 44, no. 4, pp. 1–14, 2020.

[9] J. Ma, A. Ovalle, and D. M.-k. Woodbridge, "Medhere: A smartwatch-based medication adherence monitoring system using machine learning and distributed computing," in *2018 40th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*. IEEE, 2018, pp. 4945–4948.

[10] B. Thomas. (2020) Sensorlog : Log and stream sensor data. [Online]. Available: https://apps.apple.com/us/app/sensorlog/id388014573

[11] Amazon Web Services . (2020) Amazon s3. [Online]. Available: https://aws.amazon.com/s3/

[12] Apache Spark, "Apache spark: Lightning-fast cluster computing," 2020. [Online]. Available: http://spark.apache.org

[13] Amazon Web Services . (2020) Amazon emr. [Online]. Available: https://aws.amazon.com/emr/

[14] J. S. Cramer, "The origins and development of the logit model," *Logit models from economics and other fields*, vol. 2003, pp. 1–19, 2003.

[15] L. Mason, J. Baxter, P. L. Bartlett, and M. R. Frean, "Boosting algorithms as gradient descent," in *Advances in neural information processing systems*, 2000, pp. 512–518.