Remote Medication Status Prediction for Individuals with Parkinson’s Disease using Time-series Data from Smartphones

Weijian Li*, Wei Zhu*, E. Ray Dorsey†, and Jiebo Luo*

*Department of Computer Science, University of Rochester, Rochester, NY, USA
†Center for Health + Technology and Department of Neurology, University of Rochester, Rochester, NY, USA

Email:* {whi69, wzhu15, jluo@cs.rochester.edu}, †{ray.dorsey@chet.rochester.edu}

Abstract—Medication for neurological diseases such as the Parkinson’s disease usually happens remotely away from hospitals. Such out-of-lab environments pose challenges in collecting timely and accurate health status data. Individual differences in behavioral signals collected from wearable sensors also lead to difficulties in adopting current general machine learning analysis pipelines. To address these challenges, we present a method for predicting the medication status of Parkinson’s disease patients using the public mPower dataset, which contains 62,182 remote multi-modal test records collected on smartphones from 487 patients. The proposed method shows promising results in predicting three medication statuses objectively: Before Medication (AUC=0.958), After Medication (AUC=0.958), and Another Time (AUC=0.976) by examining patient-wise historical records with the attention weights learned through a Transformer model. Our method provides an innovative way for personalized remote health sensing in a timely and objective fashion which could benefit a broad range of similar applications.

Index Terms—Remote Health Sensing, Transformer, Parkinson’s Disease

I. INTRODUCTION

Parkinson’s disease (PD) is the second most prevalent chronic neurodegenerative movement disorder disease in the world [1]. A large amount of efforts has been put into understanding [2], [3], predicting [4], [5] and providing effective treatments [6], [7] for PD. However, fewer studies have focused on remote PD medication monitoring. Similar to diabetes [8], medications for neurological disorder diseases usually need to be conducted for months or years and are usually taken remotely, e.g. in a home environment, away from hospitals. Therefore, remote medication status monitoring in a timely manner to ensure medication adherence [9], support dose frequency analysis [10], and enhance future treatment planning [11] becomes an important element in disease treatment.

To alleviate these issues, a large amount of attention has been devoted to the mobile health research [4], [5], [12]–[14]. Recently, a smartphone-based App built by Bot et al. through a research study named mPower [15] provides new opportunities for remote medication status monitoring for individuals with PD. It constructs clinical-relevant PD tests on smartphones including walking test, tapping test, voice test and memory test following the standard clinical PD measurement criteria [16]. Different from previous studies, each test is associated with a participant-reported medication point label, i.e., Immediately Before PD Medication, Just After PD Medication and Another Time (Other). The successful usage of these labeled data provides opportunities for remote medication supervision, and timely treatment plan adjustment.

Given the large quantity of accessible smartphone test records, it becomes feasible to build machine learning models to learn feature representations and further infer medication status. Although these studies [4], [15] have shown encouraging progress toward disease diagnosis, little work has focused on patient medication status prediction. In particular, our goal is to validate that the same at-home multimodal sensor signals can indeed facilitate automated monitoring of fine-grained individual behaviors (e.g., medication and response to medication) [17]. As a relatively well-defined behavior, medication time-point detection using machine learning models would be a natural first step or gateway toward achieving the ultimate goal.

To address the aforementioned issues, we introduce a framework for personalized PD medication time-point prediction with the time-series data collected in the mPower study. To be specific, we first model each patient individually by constructing patient-level records sequentially. Given several historical records whose medication status is already known, we want to predict the medication status (Before Medication, After Medication, or Another Time) for the incoming test record. A Transformer-based module then takes the tokenized and processed features as input and extracts multi-modal multi-scale feature embeddings through shuffle-and-exchange operations.

In summary, our main contributions are three-fold:

- We present a framework for PD medication status prediction, i.e. 'Before', 'After', and 'Another Time', by sampling and constructing record sequences for each patient.
- Our model extracts rich time-series features through an introduced Transformer-based module with shuffle-and-merge operations, allowing multi-scale and multi-variate information exchange.
- Comprehensive quantitative evaluations on two public datasets, detailed groupwise, and individual studies, as well as attention value visualizations demonstrate the general interpretability and applicability of our method.
II. RELATED WORK

A. Remote Medication Status Prediction

Significant efforts [4], [5], [13], [14], [18]–[20] have been devoted to remote health and medication status prediction in recent years. In the domain of PD, Bohlmann et al. [18] conducted a review indicating the huge potential in medication adherence prediction for PD. Schwad et al. [5] and Li et al. [4] build deep models for PD prediction for users with smartphone-based PD tests. Although these studies have shown encouraging progress towards disease diagnosis and medication adherence, limited work has focused on patient medication status prediction which is an important clinic-related out-of-lab indicator for medication adherence.

B. Sequential Prediction in Healthcare

Sequential prediction refers to a group of tasks that given the information of historical states, the model predicts a label for the current state. It has broad applications in data mining [21]–[23], computer vision [24], [25] and NLP [26], [27] etc. By modeling observations sequentially, temporal relationships, as well as the individual-level attributes are naturally injected as prior knowledge into the model, allowing temporal reasoning and personalized decision-making. This idea is also widely adopted in the healthcare domain [28], [29]. Recently, Luo et al. [28] introduce a time-aware attention framework for risk prediction. In their framework, a Transformer model is shown to be effective at fusing and interacting status representations among visits at different times. However, feeding all historical data into the Transformer model will lead to huge computational complexity.

C. Transformer Models in Time-series Analysis

The Transformer [30] model is originally proposed in NLP for handling long-range relationships effectively. Recently, researchers have adopted the Transformer models for processing time-series signals and reported promising results [31]–[34]. Based on the Transformer model, Li et al. [32] introduce convolutional and causal self-attention for efficient time-series forecasting. The superior results indicate Transformer’s strong ability not only for single modality modeling but also for multiple modalities or variables at the same time as well. Other researchers find that instance-to-instance relationship modeling boost global representations [35]–[38]. A common way to extract time-series signals representations is based on a set of predefined lengths which is insufficient to capture inner attributes such as periodic patterns due to length mismatch. Recent works [39], [40] suggest that token shuffling or merging alone provides enriched representation and reduce computational cost. However, the effective way to combine the benefits from both sides remains unexplored.

III. METHOD

A. Problem Definition

Given a dataset $\mathcal{X} = \{x_{1m}, ..., x_{nm}\}$, $n \in N$, $m \in M$, $t_n \in T_n$ contains records $x_{nm}$ of modality $m$, for patient $n$, at different time points $t_n$ in total number of time points $T_n$. For each patient’s record, some modality $m$ may be missing but not all of $M$. We also have the medication status labels $y_{tn} \in Y$, where $Y = \{\text{Immediately Before PD Medication}, \text{Just After PD Medication}, \text{and Another Time(Other)}\}$. Our goal is to predict $(T_n + 1)$th record’s medication status label. In other words, given the historical records of a patient, we would like to classify what the medication status $y_{tn+1}$ of the incoming record is. An illustration can be found in Figure 1.

B. Patient-level Sequence Construction

A straightforward way of handling this classification problem is to consider all the historical records as training samples, and use the corresponding labels as supervision to train a machine learning model. Then we can use the trained model to predict the $(T_n + 1)$th record in a patient-agnostic way. However, different from other classification tasks, medication status prediction is highly related to individual health and body conditions such as age, gender, disease severity, medication intake length, response to medication, etc. Besides, each person may interact with smartphones in different behavior patterns, for
example, the walking speed and gait, tapping speed and strength, even in the different test environments. These disparities could easily confound the model and lead to unsatisfying performances (please also refer to our ablative study results).

Based on these observations, we propose to tackle this problem individually by taking patient-level knowledge into account from a query-key-matching perspective with historical records as keys and incoming records as queries. As can be seen in Figure 1, for each patient \( n \in N \), his/her records \( x_{nm} \) are firstly grouped based on patient ID and are sorted based on time. After processing, record embeddings \( v_{nm}^t \in \mathbb{R}^d \) at \( t_n \in \{1, ..., T_n\} \) are merged with the corresponding medication status embeddings \( v_s \in \mathbb{R}^d \). Encoding details can be found in the next subsection.

Notice that patients may have conducted different number of tests (\( T_n \) is different). Directly computing query-key featuring matching will bring significant computation overhead especially for patients who conduct more tests than the others (\( T_n \) is larger). To overcome this issue, for each patient, we sample \( K \) records randomly from their history records at time points \( T'_n \) instead of feeding all records into the model. By adopting this few-shot training strategy, our model is not only more robust in tackling the overfitting problem, but also better handling patients with sparse records, i.e. small or different amount of test records in different modalities, which is commonly observed in real world.

### C. Tokenization and Attribute Encoding

After sampling and constructing record sequence for each patient our model takes a sequence of raw time-series signals at new sampled time points as input:

\[
X_n = \{x_{nm}^t | m \in M, t_n \in T'_n\},
\]

where each time series \( x_{nm}^t \in \mathbb{R}^{L \times 3} \) consists of 3 channels representing x, y, and z dimensional accelerometer readings. Similar to previous studies [33, 37], we flatten and chunk a time-series signal into \( P \) 1D segments \( x' \in \mathbb{R}^{P \times (S \cdot 3)} \), where \( S \) is a predefined length of the 1D segments, and \( P = \frac{T}{S} \) is the number of segments obtained. A linear layer is then used to project the chunked segments into feature vector representations:

\[
v_{nm}^T = W^T \cdot x_{nm}^T + b,
\]

where \( W \in \mathbb{R}^{(S \cdot 3) \times d} \) and \( b \in \mathbb{R}^d \).

### D. Multi-scale Transformer

The Transformer Model [30] is an encoder-decoder structure where each encoder and decoder consists of multiple layers of the attention blocks. However, due to the nature of time-series signals, feature scale and the length of time series signals are crucial in determining the representation of the patient status. The pre-defined segment length of each sequence sent into transformer limits this multi-scale representation. Directly
combining neighboring sub-sequences as new representations may generate enlarged-scale information but does not consider long-term larger-scale representations. Inspired by the ShuffleNet model [41], we present a shuffle-and-merge approach to extract multi-scale time-series representations. Each layer of the Shuffle Encoder mainly contains three parts: 1) token-shuffling, 2) token-merging, and 3) multi-head self-attention and add & Norm. Different from the ShuffleNet model [41], our method focuses on constructing multi-scale representations by shuffling time-series sequence in the temporal dimension, and proposes a merge-and-drop operation that naturally fits the need in our Transformer structure.

Firstly, token-shuffling operation shuffles the tokenized and attribute aggregated tokens from the previous module according to a random order:

$$v^n_{nm} = \text{Shuffle}\{v^n_{nm1}, v^n_{nm2}, ..., v^n_{nmp}\}$$ (4)

By doing this, we reallocate each segment leading to a set of neighbor-changed tokens. This makes previously remote tokens to become closer and previously neighboring tokens to be further apart.

Then token-merging operation takes the shuffled tokens and groups each pair of the neighboring tokens together to become a combined feature representation by concatenation and mapping:

$$v^n_{nm'} = \text{Linear}\{\text{Concat}(v^n_{nmk}, v^n_{nmk+1})\}$$ (5)

where \(\{k_i\}\) represent the new indices for the shuffled tokens. Notice that shuffling and merging operations are conducted within each source in this paper. With the help of these two operations, we obtain an additional group of tokens where each of them contains a combined representation from two previous tokens. We call this the second-order tokens. Then the multihead self-attention, add&norm same as the original Transformer model is applied to the first-order and the second-order tokens from all modalities and at all time points:

$$\text{Att}&\text{AN}(\{v^n_{n1}, ..., v^n_{nm}, v^n_{n1}, ..., v^n_{nm'}\} | m \in M, t_n \in T_n)$$ (6)

Then merged tokens are dropped. This completes one layer of the proposed Shuffle-Encoder. We reassign the shuffled tokens as the previous tokens and send them to the rest operations:

$$v_n = \text{Add}&\text{Norm}(\text{FeedForward}(v^n_{nm}))$$ (7)

The final prediction is obtained by computing a global average pooling and mapping on the output tokens of the last layer:

$$y_n = \text{MLP}\{\text{AvgPool}(v_n)\}$$ (8)

E. Training

By examining the class imbalance problem, we train our model with a weighted cross-entropy loss where the weights are assigned to each class based on the class distribution in the batch:

$$L = \frac{1}{N} \sum_{i=1}^{N} \sum_{c=1}^{C} w_c y_c \log \left( \frac{e^{v^n_{ic}}}{\sum_{j=1}^{C} e^{v^n_{ij}}} \right)$$ (9)

where \(w_c\) are classwise weights. Their values are inversely proportional to the classwise sample ratios in the batch.
We first try random guessing one of the three statuses which we examine the model performance on the mPower dataset. With PD included in our study with test records conducted was split into five subsets with similar test record sizes and class (9.4%) are memory tests. Details of the dataset information

| Method          | F1 Score | AUC  |
|-----------------|----------|------|
| Random Guess    | 0.335 ± 0.019 | 0.507 ± 0.016 |
| XGBoost         | 0.792 ± 0.021 | 0.935 ± 0.023 |
| MLP             | 0.814 ± 0.033 | 0.944 ± 0.006 |
| TCN             | 0.838 ± 0.013 | 0.948 ± 0.004 |
| Bi-LSTM         | 0.825 ± 0.027 | 0.944 ± 0.007 |
| PD              | 0.859 ± 0.021 | 0.960 ± 0.006 |
| ConvSelfAttn    | 0.866 ± 0.027 | 0.958 ± 0.006 |
| VATT            | 0.885 ± 0.027 | 0.960 ± 0.008 |
| Ours            | 0.901 ± 0.029 | 0.969 ± 0.006 |

Each group’s average observation time of the three modalities to give us the constructed sequential multimodal records for each patient.

**Other processing** We adopt a similar approach as previous studies [4], [5], [44] to extract qualified participants. Since we are more interested in predicting PD medication status, we filter out users labeled as Non-PD, and only test records that are labeled with the three target statuses. To ensure enough samples to construct sequential representation, participants who perform fewer than 6 tests in total are also not included in the study.

### V. Results

After pre-processing (sec. IV-B), there are 487 individuals with PD included in our study with test records conducted between March 09, 2015 and September 04, 2015. Among them, 189 (40.1%) are female, age ranges from [45, 86] with a mean of 63.2. 163 (41.9%), 118 (41.4%), and 125 (40.3%) of them have the last test record labeled in Another Time, Before Medication, and After Medication respectively. In total, there are 62,182 test records included and 35,417 (56.9%) are tapping tests, 20,922 (33.7%) are walking tests, and 5,843 (9.4%) are memory tests. Details of the dataset information can be found in Table I.

#### A. Performance Evaluation

**K-fold Validation** We conducted a K-fold validation (K=5) to validate our method. In detail, the dataset was first shuffled and was split into five subsets with similar test record sizes and class distributions according to the participant’s user ID. At each time, four folds were selected as training data, and the remaining one fold was used as testing data (no participant overlaps between training and testing). After obtaining the evaluation results we chose another fold as testing data and the rest four folds as training data. This process continued until all 5 folds were evaluated. We examined our method on various evaluation metrics and found promising results. For example, our method achieves an average 0.969 AUC score, and an average 0.901 F1 score for the three target medication variables. It outperformed the standard machine learning approaches (directly predicting medication status without considering patient identity and record sequences) (Table II).

**Comparison to Other Methods** We find that almost all models achieve a high F1 score of 0.792 or higher as is shown in Table II. This indicates the effectiveness of the proposed sequential modeling strategy in the medication status prediction task. A closer look at the table, we find that almost all neural network based models perform better than the non-neural network model, i.e. XGBoost. We believe higher model capacity would lead to better results in our task. Comparing to non-Transformer models, i.e. MLP, TCN and Bi-LSTM, we observe that Transformer-based models achieve higher performance. Recall that ConvSelfAttn model leverages a convolution filter to aggregate segments in local neighborhoods. This operation brings segments together and is supposed to enable locality awareness similar to our approach. However, simply merging segments unavoidably lead to more complicated representation and lost the representation by original individual segments. This indicates that smaller segments are actually important in the Transformer models. We also notice that our method achieves better performance compared to a recent domain specific method for PD classification [4]. We find that our model outperforms all previous methods including the lately proposed Transformer-based method ConvSelfAttn and VATT. In particular, our method achieves 0.918 and 0.901 in accuracy and F1 score that exceed VATT by almost 2%.

#### B. Overall Medication Pattern Visualization

Patient test record patterns could reflect test time preferences which could be used to improve remote health assessment designs for better enrollment and experience (e.g. test length per time). As shown in Figure 3, we draw line plots on
the ground-truth medication statuses as well as the predicted medication statuses across 24 hours based on all the records from participants in the testing set. For each hour, the ratios of patients for the three medication statuses are plotted with three different colors, respectively. First, we observe that most patients prefer to take a test after their medication around 10 a.m. and 15 p.m., and are less likely to do so in the early mornings or late nights. Participants prefer to conduct tests on these timeframes and label the test records as Another Time. By looking at the prediction results of our model, we find that they align with the ground-truth ratios well with slight drifts in a local range (0.04 ± 0.05). Recall that our method uses a small number of historical records (4) for each participant to construct a record sequence to make a prediction on the incoming test record. This few-shot design would be particularly useful in reducing each participant’s efforts in the data collection process in related human-subject studies.

C. Prediction Interpretation

Next, we demonstrate the model’s ability on explaining the prediction decisions quantitatively and qualitatively with the attention value. Eight box plots are plotted in Figure 4. Each of them corresponds to an incoming record from a different participant labeled with either Before Medication or After Medication. Inside each plot, all the attention values on the segments (see Methods) of four historical test records are drawn as blue boxes. The attention computation is conducted during the self-attention operation in the Transformer model. Higher attention values represent higher weights assigned to records that contain similar features. As can be seen from this figure, a stronger relationship is discovered between records with the same medication status (e.g. Day 4, Day 2, Day 4, and Day 1 in the first row; Day 4, Day 4, Day 1, and Day 1 in the second row), weaker relationships are assigned to records with different medication status. Records with the same medication status usually share similar time-series patterns (e.g. Day 5 and Day 2 in the top left corner), and records with different medication statuses are shown different patterns (e.g. Day 5 and Day 2). When there are multiple records with the same medication status, the system tends to assign higher weights to records with closer patterns (e.g. Day 5 and Day 2 in the upper right corner), or assign lower weights if a clear disagreement is observed (e.g. Day 5 and Day 3 in the lower right).

VI. CONCLUSION

In summary, we present a novel interpretable deep learning based framework for PD medication status prediction. The proposed framework models patient test records as temporal sequences and predicting current status by querying historical records. This mechanism largely increased prediction performance by modeling individual differences. The sequential record construction process requires fewer samples to achieve strong performance suitable for a variety of medical problems. Comprehensive experiments on a large public PD study dataset demonstrate the stability and explainability of the proposed methods in PD medication status prediction.

VII. ACKNOWLEDGEMENT

Research reported in this publication was supported by the National Institute Of Neurological Disorders And Stroke of the National Institutes of Health under Award Number P50NS108676. The content is solely the responsibility of the authors and does not necessarily represent the official views of the National Institutes of Health.
source-wise missing data. IEEE Transactions on Biomedical Engineering, vol. 66, no. 5, pp. 1402–1411, 2018.

[45] T. Chen and C. Guestrin, “Xgboost: A scalable tree boosting system,” in SIGKDD, 2016, pp. 785–794.

[46] M. W. Gardner and S. Dorling, “Artificial neural networks (the multilayer perceptron)—a review of applications in the atmospheric sciences,” Atmospheric environment, vol. 32, no. 14-15, pp. 2627–2636, 1998.

[47] C. Lea, M. D. Flynn, R. Vidal, A. Reiter, and G. D. Hager, “Temporal convolutional networks for action segmentation and detection,” in CVPR, 2017, pp. 156–165.

[48] A. Graves, N. Jaitly, and A.-r. Mohamed, “Hybrid speech recognition with deep bidirectional lstm,” in 2013 IEEE workshop on automatic speech recognition and understanding. IEEE, 2013, pp. 273–278.