Journal of Healthcare Informatics Research
https://doi.org/10.1007/s41666-024-00170-9

RESEARCH ARTICLE

# FENAP: Foundation Models for EMA-Derived Narrative Analysis and Prediction

**Fengxiang Zhao**[1] · **Yi Shang**[1] · **Timothy J. Trull**[2]

Received: 17 May 2024 / Revised: 21 July 2024 / Accepted: 6 August 2024


## Abstract

This study introduces FENAP, a novel approach that leverages advanced foundation models to analyze and predict human behavior using Ecological Momentary Assessment (EMA) data. FENAP serializes raw EMA inputs into structured narratives and processes them through state-of-the-art language models, such as GPT-4, to capture intricate behavioral patterns and temporal dynamics. The method then employs fine-tuned foundation models, including BERT, RoBERTa, FLAN-T5, and LLaMA-2, to predict behavioral outcomes with enhanced accuracy. Evaluations demonstrate the superiority of FENAP over traditional machine learning models, ensemble methods, and deep neural networks, with FLAN-T5 using LoRA fine-tuning achieving the lowest root mean squared error of 1.476 for regression tasks and LLaMA-2 LoRA obtaining the highest F1 score of 62.95 for classification tasks. These findings highlight the effectiveness of leveraging advanced foundation models and efficient fine-tuning techniques for capturing complex patterns and relationships in EMA-derived narratives. FENAP establishes a foundational framework for future research and practical applications in dynamically understanding and predicting human behavior using EMA data, with potential implications across various domains, such as health psychology, behavioral intervention, and real-time monitoring.

**Keywords** Ecological momentary assessment · Foundation models · Human behavior prediction · Large language models · Data analysis · Natural language processing

✉ Yi Shang
shangy@missouri.edu

✉ Timothy J. Trull
trullt@missouri.edu

Fengxiang Zhao
fzfmx@missouri.edu

1 EECS, University of Missouri-Columbia, Columbia, MO, USA

2 Department of Psychology, University of Missouri-Columbia, Columbia, MO, USA

Published online: 20 August 2024

## 1 Introduction

Ecological Momentary Assessments (EMAs) have emerged as a powerful tool for capturing real-time data on human behavior and experiences in natural settings. These assessments generate rich data that provide valuable insights into various aspects of individuals' lives, such as their thoughts, feelings, activities, and environmental contexts [1]. However, the complex and high-dimensional nature of EMA data presents significant challenges for traditional data analysis methods, limiting their ability to fully exploit the potential of these datasets for understanding and predicting human behavior.

Recent advancements in machine learning (ML) and artificial intelligence (AI), particularly the development of large language models (LLMs) like GPT variants, have opened up new possibilities for effectively processing and analyzing complex, unstructured data. These models have demonstrated remarkable capabilities in natural language understanding, generation, and reasoning, making them well-suited for extracting meaningful insights from diverse data sources, including EMA-derived narratives.

In this study, we propose a novel methodology called `FENAP` (Foundation Models for Comprehensive Analysis and Prediction of EMA-derived Narratives), which leverages the power of advanced foundation models to analyze and predict human behavior using EMA data. Our approach involves three key contributions:

1. **Methodology proposal:** We introduce a novel approach that employs large foundation models to interpret and analyze EMA data effectively. By transforming raw EMA inputs into structured narratives and processing them through state-of-the-art language models, `FENAP` captures intricate behavioral patterns and temporal dynamics, enabling more accurate and comprehensive analysis of human behavior.
2. **Method implementation:** To demonstrate the practical application of our proposed methodology, we implement `FENAP` using a carefully curated dataset from the UT1000 Project [2], which encompasses a wide range of behavioral and environmental data collected from a large cohort of participants. Specifically, we utilize a diverse set of state-of-the-art foundation models to capture the intricate patterns and dynamics within the EMA-derived narratives. Furthermore, we explore different fine-tuning approaches, such as training all parameters, training only the last layer, and using the LoRA technique, to optimize the adaptation of these foundation models to the specific task of processing and predicting behavioral outcomes from EMA data.
3. **Performance evaluation:** We conduct rigorous experiments to evaluate the effectiveness of `FENAP` in predicting behavioral outcomes and compare its performance against a wide range of baseline methods. These baselines include traditional machine learning models (e.g., linear regression, decision tree, random forest, SVM), ensemble methods (e.g., AdaBoost, XGBoost, LightGBM, CatBoost), and state-of-the-art deep neural networks specifically designed for tabular data (e.g., MLP, DeepFM, TabNet). Our experimental results show that `FENAP` achieves around 11% reduction in root mean squared error (RMSE) in regression task and around 13% increment in F1 score in classification task compared to the

best-performing baseline method, highlighting the significant improvement in predictive accuracy offered by our approach.

By integrating cutting-edge AI techniques with the rich data generated by EMAs, FENAP offers a powerful and innovative approach to dynamically understanding and predicting human behavior. This study establishes a foundational framework for future research and practical applications, showcasing the immense potential of leveraging advanced foundation models to unlock new insights from complex behavioral datasets.

The remainder of this paper is organized as follows: Sect. 2 provides an overview of related works in the fields of EMA data analysis and foundation models. Section 3 presents the detailed methodology of FENAP, including the problem formulation and implementation details. Section 4 describes the experimental setup, including the dataset, baselines, and evaluation metrics. Section 5 presents the results and analysis of our experiments, comparing the performance of FENAP against various baseline methods. Finally, Sect. 6 discusses the implications of our findings, limitations, and future research directions.

## 2 Related Works

### 2.1 Machine Learning Approaches for EMA Data Analysis

Machine learning methods have been widely applied to analyze and interpret EMA data, leveraging their ability to capture complex patterns and relationships in structured data. Common approaches include linear regression [3], decision trees, random forests, and support vector machines, which have been successfully employed in various EMA studies, such as mood and behavior assessments [4–7]. These methods are particularly valuable for their effectiveness and simplicity in handling structured EMA data.

Researchers have also developed novel machine learning algorithms specifically tailored for EMA data analysis. For instance, Spanakis et al. [8] introduced the Bagged Boosted Trees (BBT) algorithm, which enhances decision tree models for EMA data by incorporating over/under sampling methods to improve classification accuracy in complex, structured data environments. Additionally, Aminikhanghahi et al. [9] presented a machine learning-based method for context-aware EMA prompting, significantly increasing participant engagement and response rates by timing EMA prompts during transitions in participant activity.

### 2.2 Foundation Models and Their Potential in Behavioral Research

Foundation models, such as BERT, GPT, and other pre-trained language models, have revolutionized natural language processing by demonstrating superior performance across a wide range of tasks. These models are initially trained on large text corpora to develop a comprehensive understanding of language, which can then be fine-tuned to perform specific tasks with remarkable effectiveness. The adaptability of foundation models across various domains is largely due to their architecture and the comprehensive nature of their training data, allowing them to be applied to tasks in different

fields beyond language processing, such as healthcare, finance, and legal analysis [10, 11].

One of the major advantages of foundation models is their ability to extract meaningful insights from raw data, significantly reducing the need for extensive preprocessing. This capability enables researchers to deploy these models directly tabular, obtaining high-quality outputs [12–15]. As a result, foundation models hold significant potential in analyzing and predicting through textual data. Their ability to understand and process natural language at a high level allows them to identify patterns and insights that can inform behavioral interventions and psychological studies by integrating tabular data with natural language processing; these models can enhance the interpretation of structured data [15]. This integration allows for analysis and prediction, broadening the scope of research applications. For example, McMaster et al. [16] explore methods for adapting pretrained language models to tackle tabular prediction challenges within electronic health records, demonstrating enhanced model performance on clinical data tasks.

### 2.3 Gaps in Current Research and Motivation for FENAP

Despite the growing popularity of EMA in behavioral research, existing approaches have limitations in fully exploiting the potential of EMA data due to their inability to capture the complex and high-dimensional nature of these datasets. To address this gap, there is a need for a method that leverages advanced foundation models, such as BERT, GPT, and other pre-trained language models, for EMA data analysis. The proposed FENAP approach aims to fill this gap by transforming raw EMA inputs into structured narratives and processing them through state-of-the-art language models, enabling the capture of intricate behavioral patterns and temporal dynamics that may be overlooked by traditional approaches. By combining EMA-derived narratives with foundation models, FENAP can fully utilize the rich and diverse data collected through EMAs, identify complex patterns and relationships within the data, and make accurate predictions about behavioral outcomes, opening up new avenues for research and practical applications in behavioral science.

## 3 Methodology

In this section, we outline the methodology for our proposed approach, which leverages large foundation models for dynamic interpretation and analysis of EMA data. We detail the conceptual structure, formulation, and implementation stages of this method.

### 3.1 Overview

Our methodology for processing and predicting behavioral outcomes based on EMA data consists of a 3-step approach, as demonstrated in Fig. 1.

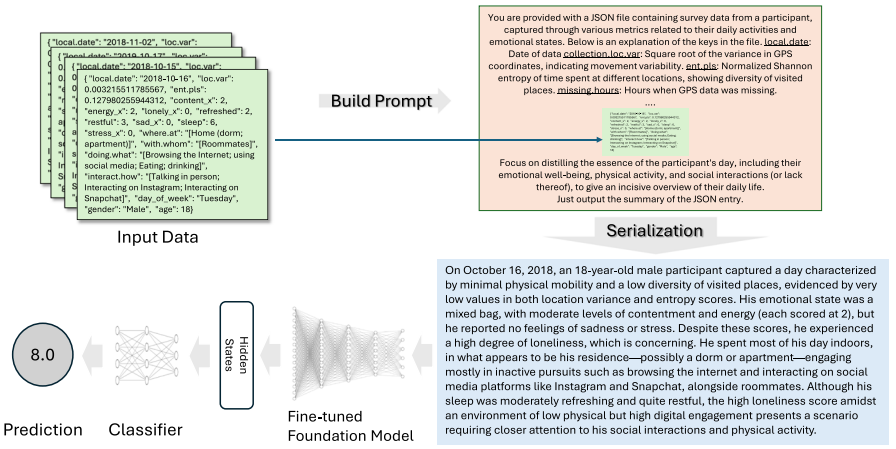


**Fig. 1** Overview of the FENAP pipeline for processing and predicting behavioral outcomes from EMA data. First, prompt is built based on survey data with the pre-defined template. Second, data is then serialized into a descriptive natural language text using Large Language Model like GPT-4. Then, we use data narrative to fine-tune the foundation models, such as LLaMA. Last, the fine-tuned model is applied to a small classifier, like a feed-forward network to make the prediction

### 3.1.1 Prompt Generation

To convert raw EMA data into a structured narrative form, a predefined template that includes a brief explanation of each JSON key, the JSON data itself, and an instruction to summarize the participant's behavior, as exemplified at Code Listing 1, was used. This template provides a brief explanation for each JSON key, allowing the model to understand the meaning of the data fields without including the full descriptions. The instruction to summarize the participant's day remains the same, focusing on their emotional well-being, physical activity, and social interactions.

### 3.1.2 Serialization

The preformatted instructions are input into an advanced large language model, such as GPT-4. These models generate coherent and comprehensive narrative profiles that integrate data inputs cohesively, ensuring that all relevant information is maintained in a usable and interpretable format.

### 3.1.3 Predictive Modeling via Fine-Tuning

The narratives produced in previous step serve as the input for fine-tuning larger foundation models like BERT [17], RoBERTa [18], FLAN-T5 [19], and LLaMA-2 [20]. The refined models analyze the narrative profiles to learn complex behavioral patterns and contextual cues inherent in the EMA data. A small classifier, typically A feed-forward neural network, is used to predict the value based on the hidden states derived from the large foundation model.

This structured approach permits a nuanced analysis of layered EMA data, enabling more accurate predictions and insights into human behavior and health metrics in naturalistic settings. The subsequent sections will delve into the specific details of each step and discuss the implementation process using the dataset derived from the UT1000 Project [2].

### 3.2 Problem Definition

Consider an EMA dataset $\mathcal{D} = \{\mathcal{X}, \mathbf{y}\}$, where $\mathcal{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_m\} \in \mathbb{R}^{m \times n}$ represents $m$ data instances, each consisting of $n$ features. Here, $\mathbf{y} \in \mathbb{R}^m$ denotes the target variable, which can be either continuous (for regression tasks) or categorical (for classification tasks), representing various behavioral and health metrics. Columns in $\mathcal{X}$, $k_j$, $j = 1, \cdots, n$, represent the features.

#### 3.2.1 Prompt Generation

As detailed in Sect. 3.1.1, we convert each instance of EMA data $\mathbf{x}_i = [x_i^1, \ldots, x_i^n]$ into a structured prompt $\mathbf{p}_i$. The prompt is generated using a predefined template $\mathcal{T}$, where the JSON object $\mathbf{j}_i$ containing the EMA data is inserted into designated placeholders within the template:

$$\mathbf{p}_i = \mathcal{T}(\mathbf{j}_i), \quad i = 1, \ldots, m \tag{1}$$

The JSON object $\mathbf{j}_i$ is constructed by combining each feature name $k_j$ with its corresponding feature description $d_j$:

$$\mathbf{j}_i = \{k_1 : d_1, \ldots, k_n : d_n\}, \quad i = 1, \ldots, m \tag{2}$$

From these transformations, we generate a set of prompts $\mathcal{P} = \{\mathbf{p}_1, \ldots, \mathbf{p}_m\}$ containing $m$ entries, with each entry representing a complete prompt that includes both the EMA data and the feature name-description pairs.

#### 3.2.2 Narrative Construction

Using the set of prompts $\mathcal{P}$ generated in the previous step, we feed each prompt $\mathbf{p}_i$ into a high-capacity language model $\mathcal{L}$ like GPT-4 via the OpenAI API:

$$\mathbf{n}_i = \mathcal{L}(\mathbf{p}_i), \quad i = 1, \ldots, m \tag{3}$$

The language model processes these prompts and generates a corresponding set of detailed narrative profiles $\mathcal{N} = \{\mathbf{n}_1, \ldots, \mathbf{n}_m\}$, where each narrative profile $\mathbf{n}_i$ provides a comprehensive and contextual representation of the original EMA data instance $\mathbf{x}_i$.

### 3.2.3 Predictive Modeling

As described in Sect. 3.1.3, foundational models such as BERT and LLaMA are fine-tuned with the narrative profiles $\mathcal{N}$ derived from the previous step. These narratives are first tokenized using the official tokenizer of the respective models, producing a tokenized matrix $\mathbf{T} \in \mathbb{R}^{m \times t} = \{t_i\}_{i=1}^{m}$ where $t$ is the number of tokens per input narrative.

Our method can be applied to both regression and classification tasks. For regression tasks, we append a regression head $\mathcal{R} : \mathbb{R}^{t \times d} \rightarrow \mathbb{R}$ to each foundational model, where $d$ is the dimension of the latent representations from the foundational model. This regression head calculates the predicted continuous outcomes $\hat{\mathbf{y}}$ from the latent embeddings $\mathbf{h}$ as follows:

$$\mathbf{h} = \mathcal{R}(\mathbf{t}) \tag{4}$$

For classification tasks, we append a classification head $\mathcal{C} : \mathbb{R}^{t \times d} \rightarrow \mathbb{R}^{c}$ to each foundational model, where $c$ is the number of classes. This classification head calculates the predicted class probabilities $\hat{\mathbf{y}}$ from the latent embeddings $\mathbf{h}$ as follows:

$$\mathbf{h} = \mathcal{C}(\mathbf{t}) \tag{5}$$

Depending on the architecture of the foundational model, the approach to deriving summary representations $\mathbf{h}_{\text{reg}}$ or $\mathbf{h}_{\text{class}}$ may vary. For bidirectional models like BERT, we can average the latent representations, whereas for autoregressive models like GPT, we might utilize the representation of the last token.

$$\hat{y} = \mathcal{R}(\mathbf{h}_{\text{reg}}) \tag{6}$$

$$\hat{y} = \mathcal{C}(\mathbf{h}_{\text{class}}) \tag{7}$$

For regression tasks, the performance of the model is evaluated using the Root Mean Squared Error (RMSE) between the predicted outcomes $\hat{\mathbf{y}}$ and the true values $\mathbf{y}$, defined as

$$\text{RMSE} = \sqrt{\frac{1}{m}\sum_{i=1}^{m}(\hat{y}_i - y_i)^2} \tag{8}$$

For classification tasks, the performance of the model is evaluated using either the Binary Cross-Entropy (BCE) or Categorical Cross-Entropy (CE) loss, depending on the number of classes in the problem.

When dealing with binary classification problems (i.e., two classes), we use the Binary Cross-Entropy (BCE) loss, defined as

$$\text{BCE} = -\frac{1}{m}\sum_{i=1}^{m}[y_i \log(\hat{y}_i) + (1 - y_i)\log(1 - \hat{y}_i)] \tag{9}$$

where $y_i \in \{0, 1\}$ is the true binary label and $\hat{y}_i \in [0, 1]$ is the predicted probability for the positive class.

For multi-class classification problems (i.e., more than two classes), we use the Categorical Cross-Entropy (CE) loss, defined as

$$\text{CE} = -\frac{1}{m} \sum_{i=1}^{m} \sum_{c=1}^{C} y_{i,c} \log(\hat{y}_{i,c}) \tag{10}$$

where $C$ is the number of classes, $y_{i,c} \in \{0, 1\}$ is the true binary indicator of whether class $c$ is the correct classification for instance $i$, and $\hat{y}_{i,c} \in [0, 1]$ is the predicted probability that instance $i$ belongs to class $c$.

In both cases, the loss functions are averaged over all instances in the dataset.

## 4 Experiment

This section details our experimental setup, including Dataset Preparation (Sect. 4.1), Experiment Setup (Sect. 4.2), Chosen Foundation Models for `FENAP` (Sect. 4.3), Baselines (Sect. 4.4), Implementation Specifics (Sect. 4.5), and the Evaluation Metrics Used (Sect. 4.6).

### 4.1 Dataset

#### 4.1.1 Summary

The study utilizes a rich, multi-modal dataset from the UT1000 Project [2], a large-scale data collection study conducted at the University of Texas at Austin. The project aimed to measure various aspects of health, behavior, and home environment among a cohort of 1584 participants (62% female) using a wide range of technologies and methods, including traditional surveys, swabbing, EMAs, smartphone sensing, wearable trackers, and environmental sensors. The study was conducted in two deployments, one in the Fall of 2018 and the other in the Spring of 2019, each lasting 3 weeks.

For our experiment, we applied 4 interrelated components of the UT1000 Project dataset:

1. **GPS Data Features** [21]**:** This subset encapsulates the geographical movement patterns of participants, processed to safeguard privacy. It includes calculated features such as the variance in daily GPS coordinates and the entropy of time spent at various significant locations. These metrics are essential for understanding spatial behaviors and their potential impacts on health and lifestyle.
2. **Ecological Momentary Assessment (EMA) Data** [22]**:** EMAs were administered using the Beiwe mobile application at regularly scheduled times throughout each day. The EMA questions covered four categories: sleep, momentary context, momentary well-being, and an audio question. These data provide a dynamic snapshot of the participants' mental health, daily activities, and behaviors.

3. **Accelerometer** [23]**:** The dataset comprises accelerometer data from the Beiwe mobile app, detailing daily acceleration magnitudes across three axes (x, y, z) measured in gravity units (9.8 $m/s^2$). Features include daily statistical measures like mean, standard deviation, minimum, maximum, and root mean square of successive differences (rmssd). It also includes "acc.still," indicating periods of inactivity, and "acc.complete," which reflects the completeness of daily data coverage.
4. **Demographic and Basic Information** [24]**:** Basic demographic details such as age, gender, and the period of study participation are included to support demographic-specific analyses and to ensure the applicability of findings across different population subgroups. This information is crucial for contextualizing the behavioral and environmental data within demographic trends.

#### 4.1.2 Data Preprocessing

To prepare the UT1000 Project dataset for subsequent analysis and modeling, our primary preprocessing step focused on consolidating diverse data types into a unified tabular format. Specifically, we performed an inner join operation on GPS data, EMA responses, and demographic details based on each participant's ID and the respective date. This integration process created a tabular dataset that aligns locational, temporal, and survey-derived information along with key demographic attributes for every participant for each day.

For the baseline models, we further processed the dataset by converting string features into one-hot encoded representations. This step is necessary because traditional machine learning models cannot directly handle string inputs. One-hot encoding creates binary vectors for each unique string value, allowing the models to effectively utilize the categorical information. Examples of the one-hot encoded inputs for the baseline models are detailed in Section 4.4 and can be found in Code Listing 4.

In contrast, the `FENAP` implementation does not require one-hot encoding of string features. The `FENAP` can directly process and understand textual inputs, eliminating the need for manual feature engineering of categorical variables. This capability simplifies the data preprocessing pipeline and allows for more natural representation of the data. Examples of the string inputs for the `FENAP` implementation are provided in Section 4.3 and can be found in Code Listing 3.

### 4.2 Experimental Setup

To validate the efficacy of `FENAP`, we designed two experiments targeting different types of outcomes based on the integrated datasets:

1. **Regression experiment:** This experiment involves a predictive model that combines GPS data, demographic details, and EMA responses. The model predicts the *sleep* attribute, quantified as the number of sleep hours, represented as a floating-point number. The objective is to determine the accuracy of the model in forecasting sleep durations from daily behavioral patterns and demographic data.
2. **Classification experiment:** In this setup, we utilize accelerometer data, demographic information, and EMA responses to predict the *content* attribute, an integer

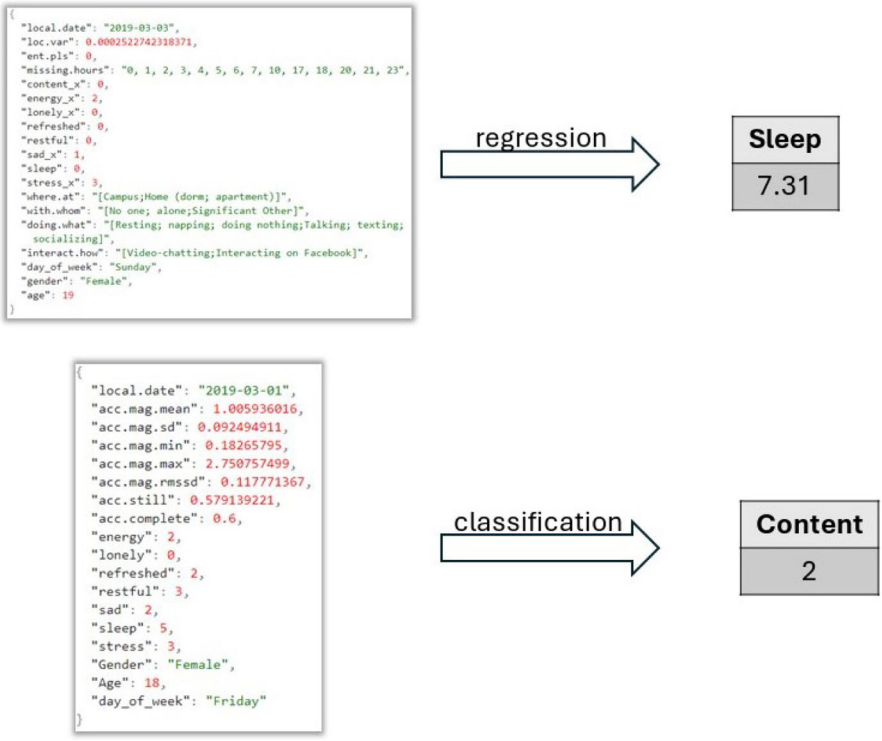


**Fig. 2** Demonstration for two experiments. The upper part of the graph focuses on the regression task of predicting sleep duration, while the lower part highlights the classification task aimed at determining subjective contentment scores based on various input metrics

that ranges from 0 to 3, representing varying levels of contentment recorded each day. The objective is to evaluate how effectively FENAP can classify input into these contentment categories.

Figure 2 demonstrates the two experiments designed for the evaluation of FENAP.

### 4.3 Implementation of FENAP

We utilized the LLaMA-2-70b-chat-hf[1] model to generate a serialized narrative description of the integrated tabular data, as detailed in Sect. 3.1.2. The prompt used is demonstrated in Code Listing 1 and remains consistent for processing all the data. We mostly use the default parameters by setting a maximum token limit of 1024, a temperature of 0.7, top_p at 0.9, top_k at 0, and a repetition penalty of 1. The system prompt we utilized is shown in Code Listing 2.

A set of advanced pretrained foundation models was employed for the implementation of FENAP in Sect. 3.1.3.

[1] https://huggingface.co/meta-llama/Llama-2-70b-chat-hf

- **BERT (Bidirectional Encoder Representations from Transformers)** [17]**:** A powerful transformer-based model that learns contextual representations of text by jointly conditioning on both left and right context. We employ two fine-tuning approaches for BERT: (1) utilizing the full parameter version of the model, allowing all parameters to be updated during fine-tuning, and (2) training only the last layer while keeping the earlier layers fixed. These approaches enable us to compare the effectiveness of fine-tuning the entire model versus fine-tuning only the last layer for adapting BERT to the specific task of processing EMA-derived narratives.
- **RoBERTa (Robustly Optimized BERT Pretraining Approach)** [18]**:** An optimized version of BERT trained on a larger corpus with improved training techniques, resulting in enhanced performance on various natural language processing tasks. Similar to BERT, we employ two fine-tuning approaches for RoBERTa: (1) utilizing the full parameter version of the model, allowing all parameters to be updated during fine-tuning, and (2) training only the last layer while keeping the earlier layers fixed. These approaches allow us to assess the effectiveness of fine-tuning the entire model compared to fine-tuning only the last layer, benefiting from RoBERTa's optimized pretraining while adapting it for our specific application.
- **PubMedBERT (PubMed Bidirectional Encoder Representations from Transformers)** [25]**:** Customized for biomedical text, this BERT variant is pre-trained exclusively on the vast collection of PubMed articles, ensuring relevance and enhanced performance in biomedical applications. We also explore two fine-tuning methods: full model updating and last layer only adjustments.
- **BioBERT (Biomedical BERT)** [26]**:** This version refines BERT through pre-training on extensive biomedical literature, including PubMed abstracts and PMC full texts, aiming to capture the nuances of biomedical language better. For fine-tuning, we compare a full parameter update against last layer only modifications.
- **FLAN-T5 (Fine-tuned Language Models with Adaptive Pretraining on Task-specific Data)** [19]**:** A variant of the T5 model [27] fine-tuned on a diverse set of tasks using adaptive pretraining, enabling effective knowledge transfer across different tasks and domains. We employ two fine-tuning approaches for FLAN-T5: (1) adapting the LoRA (Low-Rank Adaptation) [28] technique, which efficiently fine-tunes the entire model by training a small set of auxiliary parameters while keeping the original model parameters fixed, and (2) fine-tuning only the last layer of the model. These approaches allow us to compare the effectiveness of fine-tuning the entire model using LoRA versus fine-tuning only the last layer for our specific application.
- **LLaMA-2 (Large Language Model Meta AI, version 2)** [20]**:** The second version of Meta AI's autoregressive large language models, demonstrating impressive performance on various natural language processing benchmarks. Similar to FLAN-T5, we employ two fine-tuning approaches for LLaMA-2: (1) adapting the LoRA technique to efficiently fine-tune the entire model and (2) fine-tuning only the last layer of the model. These approaches enable us to assess the effectiveness of fine-tuning the entire model using LoRA compared to fine-tuning only the last layer, allowing the models to specialize in processing EMA-derived narratives while maintaining their pretrained knowledge.

### 4.4 Baselines

To benchmark the performance of FENAP, we compare it against a variety of established baselines well-suited for classification and regression tasks, primarily focusing on tree-based models and advanced neural network architectures designed for tabular data.

**Traditional Machine Learning Models** For our baseline comparisons, we implement several traditional machine learning models known for their effectiveness in various predictive tasks:

- **Decision Tree** [29]**:** A model that captures complex, non-linear relationships through a hierarchical structure of decisions.
- **Random Forest** [30]**:** An ensemble method that combines multiple decision trees to improve prediction accuracy and control over-fitting.
- **Support Vector Machine (SVM)** [31]**:** A model that finds the optimal hyperplane within a high-dimensional space to predict continuous values, providing robustness especially in cases with clear margins of separation in the data.

These models are chosen for their diverse approaches to regression, offering a broad comparison spectrum for evaluating the performance of our methodology.

**Ensemble and Boosting Methods** In addition to traditional regression models, we incorporate a suite of advanced ensemble and boosting techniques to set comprehensive baselines for our experiments:

- **AdaBoost** [32]**:** An ensemble method that combines multiple weak learners (typically decision trees) to form a strong prediction model by focusing iteratively on incorrectly predicted instances to improve model accuracy.
- **XGBoost** [33]**:** A gradient boosting framework known for its efficiency and effectiveness in handling large-scale data, optimizing both computational speed and model performance.
- **LightGBM** [34]**:** Another gradient boosting framework renowned for its fast training speeds and efficient handling of large data sets, with a focus on tree-based learning algorithms.
- **CatBoost** [35]**:** A gradient boosting algorithm that robustly handles categorical features directly and employs sophisticated techniques to reduce overfitting and increase prediction stability.

**Deep Learning Models** To explore the efficacy of deep learning techniques in regression tasks involving EMA data, we have selected several neural network architectures designed specifically for handling complex and high-dimensional tabular datasets:

- **MLP (Multilayer Perceptron)** [36]**:** A fundamental neural network architecture used for regression tasks, capable of capturing non-linear relationships in the data through multiple layers of interconnected neurons.

- **DeepFM (Deep Factorization Machine)** [37]**:** A model that combines the strengths of factorization machines for learning feature interactions and deep learning for capturing complex non-linear relationships, providing a powerful approach for handling high-dimensional and sparse data.
- **TabNet** [38]**:** An innovative neural network architecture that employs sequential attention mechanisms to selectively choose which features to process at each decision step, offering interpretable decisions and high performance in structured data tasks.

### 4.5 Implementation Details

For traditional Machine Learning and ensemble methods, we use the scikit-learn, XGBoost, LightGBM, and CatBoost libraries. The models included in our experiments are Decision Tree, Random Forest, Support Vector Regression (SVR), AdaBoost, XGBoost, LightGBM, and CatBoost. We employ a consistent preprocessing pipeline across all models to ensure a fair comparison, using the `StandardScaler` from scikit-learn to standardize the input features. The models are then trained using their respective libraries: `DecisionTreeRegressor`, `RandomForestRegressor`, `SVR`, and `AdaBoostRegressor` from scikit-learn; `XGBRegressor`, `LGBMRegressor`, and `CatBoostRegressor` from their corresponding libraries. The hyperparameters for each model are set to their default values, with the exception of the random state being fixed to 42 for reproducibility in the tree-based models (Decision Tree, Random Forest, AdaBoost, XGBoost, LightGBM, and CatBoost).

Regarding the deep learning models, the Multilayer Perceptron (MLP) is structured with an input layer dimension determined by the number of features in the training data (e.g., 100 features), followed by three hidden layers with 128, 64, and 32 neurons, respectively, and a ReLU activation function to introduce non-linearity. The output layer is set to 1 for regression tasks. We employ the Adam optimizer with a learning rate of 0.001 and use Mean Squared Error (MSE) as the loss function. The batch size is set at 64, and training is conducted for up to 100 epochs with early stopping if validation loss fails to improve for 10 consecutive epochs. For DeepFM, the model combines a Factorization Machine (FM) handling first and second-order feature interactions with a Deep Neural Network (DNN) that includes two hidden layers (128 and 64 neurons) with ReLU activation. The feature and factor count match the input data

**Table 1** Cohort distribution of the training, test, and validation sets

| Metric | Training | Test | Validation |
|---|---|---|---|
| Count | 18038 | 2254 | 2255 |
| 0 (%) | 5.33 | 5.72 | 5.41 |
| 1 (%) | 33.51 | 32.70 | 33.17 |
| 2 (%) | 39.26 | 39.57 | 40.58 |
| 3 (%) | 21.90 | 22.01 | 20.84 |

Rows 0, 1, 2, and 3 in metric represent the percentage of each content category

**Table 2** Statistics of the *sleep* (number of self-reported hours slept) column in dataset

| Statistic | Training set | Test set | Validation set |
|---|---|---|---|
| Count | 18550 | 2319 | 2319 |
| Mean | 6.19 | 6.19 | 6.21 |
| Standard deviation | 1.73 | 1.72 | 1.74 |
| Min | 0.00 | 0.00 | 0.00 |
| 25th percentile | 5.00 | 5.00 | 5.00 |
| Median (50th percentile) | 6.00 | 6.00 | 6.00 |
| 75th percentile | 7.00 | 7.00 | 7.00 |
| Max | 12.00 | 12.00 | 12.00 |

specifications, and it shares the same optimizer, loss, and training regimen as the MLP. The TabNet model utilizes the `TabNetRegressor` with default settings, trained on MinMaxScaled inputs for a maximum of 100 epochs, and employs early stopping based on validation performance improvements. The initial learning rate is set at 0.02, adjustable via the built-in scheduler. This detailed parameterization ensures clarity and replicability of our model setups.

For foundation models, we use Hugging Face Transformers library to train BERT and RoBERTa with a batch size of 256 and a learning rate of 5e-6. We use a linear learning rate scheduler and set the max number of training epochs to 100. For FLAN-T5, we used `peft` [39] library. We set the rank to 16 and the scaling factor to 32 with a dropout probability of 0.05. LLaMa-Factory [40] was used to train on the 4-bit quantized Llama-2-7b model from UnslothAI,[2] and the batch size is set to 2, with a gradient accumulation step of 4, along with a warmup ratio of 0.1 for LoRA. The model is trained for 3 epochs. The learning rate is set to 5e-5, and the gradient norm is clipped to 1.0.

We train all models on NVIDIA A100 GPUs using mixed precision to optimize training efficiency and memory usage. The AdamW optimizer is employed with fine-tuned hyperparameters for optimal convergence. The training process is divided into training (80%), validation (10%), and testing (10%) sets, ensuring comprehensive evaluation and fine-tuning on diverse data splits, and Tables 1 and 2 show the cohort distribution and statistics of the "sleep" column, respectively. Multiple training cycles with different random seed settings are conducted to ensure reproducibility and robustness of the evaluated metrics.

### 4.6 Evaluation Metrics

For the classification tasks, which involve categorizing data into predefined classes, we use the F1 score as the primary metric. The F1 score is the harmonic mean of precision and recall, and it provides a single measure of a test's accuracy by balancing

[2] https://huggingface.co/unsloth/llama-2-7b

the trade-off between precision and recall. A higher F1 score indicates more accurate predictions. The formula for the F1 score is

$$F1 = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}$$

For regression tasks where the goal is to predict a continuous variable, we utilize the Root Mean Squared Error (RMSE) in (8), which quantifies the average differences between the predicted values and the actual observations, with lower scores reflecting higher accuracy in predictions.

## 5 Results and Analysis

Table 3 presents the performance comparison of various methods for predicting human behavior from EMA-derived narratives, including the root mean squared error (RMSE) for regression tasks and the estimated F1 scores for classification tasks. The methods are categorized into four groups: traditional machine learning methods, ensemble methods, deep neural networks, and our proposed approach, FENAP.

For the regression task, among the traditional machine learning methods, Support Vector Machine (SVM) achieves the lowest RMSE of 1.669, outperforming decision trees and random forests. In the ensemble methods category, LightGBM obtains the best performance with an RMSE of 1.661, slightly surpassing other ensemble techniques. Moving on to deep neural networks, TabNet emerges as the top performer with an RMSE of 1.656, outperforming MLP and DeepFM. For BERT variants, BioBERT with all tokens demonstrates exceptional performance with the lowest RMSE of 1.541.

Best results for the regression task are obtained by our proposed approach, FENAP. For BERT variants, BioBERT with all tokens demonstrates exceptional performance with the lowest RMSE of 1.541, outperformed other methods. Among the FENAP variants, FLAN-T5 with LoRA achieves the lowest RMSE of 1.476, setting a new state-of-the-art performance on this task. This demonstrates the effectiveness of fine-tuning large-scale language models using the LoRA technique, which efficiently adapts the model to the specific domain while preserving its pretrained knowledge.

For the classification task, the F1 scores provide insights into the methods' performance in correctly identifying the target classes. Among the traditional methods, Random Forest achieves the highest F1 score of 55.53, outperforming Decision Tree and SVM. In the ensemble methods category, CatBoost obtains the best performance with an F1 score of 56.27, slightly surpassing other ensemble techniques. Among the deep learning methods, TabNet emerges as the top performer with an F1 score of 55.58.

Once again, FENAP demonstrates superior performance in the classification task. Although the BERT variants perform comparably to baseline models, LLaMA-2 LoRA achieving the highest F1 score of 62.95, followed closely by FLAN-T5 LoRA with an F1 score of 61.17. These results highlight the effectiveness of fine-tuning large-scale language models for classification tasks, capturing the complex patterns and relationships present in the EMA-derived narratives.

Notably, the performance differences between the LoRA and Last variations, such as FLAN-T5-XXL Last at 61.06 and LLaMA-2 Last at 62.56 for classification task, are

**Table 3** Performance comparison of various methods for predicting human behavior from EMA-derived dataset, including RMSE for regression task and F1 scores for classficitaion task

| Model | RMSE (↓) | F1 score (↑) |
|---|---|---|
| *Traditional methods* | | |
| Decision Tree | 2.141 | 46.05 |
| Random Forest | 1.755 | **55.53** |
| SVM | **1.669** | 22.62 |
| *Ensemble methods* | | |
| AdaBoost | 1.670 | 53.94 |
| XGBoost | 1.691 | 55.09 |
| LightGBM | **1.661** | 55.31 |
| CatBoost | 1.666 | **56.27** |
| *Deep Learning methods* | | |
| MLP | 1.725 | 54.06 |
| DeepFM | 1.877 | 55.29 |
| TabNet | **1.656** | **55.58** |
| *FENAP* (BERT variants) | | |
| BERT All | 1.599 | 54.88 |
| BERT Last | 1.758 | 51.70 |
| RoBERTa All | 1.588 | 55.53 |
| RoBERTa Last | 1.691 | 50.40 |
| PubMedBERT All | 1.568 | 56.21 |
| PubMedBERT Last | 1.624 | 51.97 |
| BioBERT All | **1.541** | **57.15** |
| BioBERT Last | 1.772 | 52.83 |
| *FENAP* (large foundation models) | | |
| FLAN-T5-XXL LoRA | **1.476** | 61.17 |
| FLAN-T5-XXL Last | 1.484 | 61.06 |
| LLaMA-2 LoRA | 1.507 | **62.95** |
| LLaMA-2 Last | 1.529 | 62.56 |

The best results are highlighted accordingly

modest, indicating that the benefits of the LoRA technique are marginal. These results underscore the efficacy of fine-tuning large-scale language models for classification tasks, particularly their capacity to discern complex patterns and relationships within EMA-derived narratives.

In summary, our proposed approach, FENAP, significantly outperforms traditional machine learning methods, ensemble methods, and deep neural networks in both regression and classification tasks for predicting human behavior from EMA-derived narratives. The superior performance of FENAP highlights the importance of leveraging advanced pretrained foundation models and efficient fine-tuning techniques, such as LoRA, to capture the complex patterns and relationships present in the data. These findings establish FENAP as a promising framework for future research and practical applications in dynamically understanding and predicting human behavior using EMA data.

## 6 Conclusion and Discussion

In this study, we introduced FENAP, a pioneering approach that harnesses advanced foundation models to analyze and predict human behavior using Ecological Momentary Assessment (EMA) data. By converting raw EMA inputs into structured narratives and processing them through state-of-the-art language models like GPT-4 and FLAN-T5, FENAP captures intricate behavioral patterns and improves prediction accuracy. Among various methods tested, FLAN-T5 with LoRA fine-tuning showed the most promise, achieving the lowest mean squared error (MSE) and underscoring the power of combining large-scale pre-trained models and fine-tuning techniques for complex data interpretation.

The success of FENAP provides a robust framework for further research and practical applications in fields like health psychology and real-time behavioral intervention. However, the study's reliance on the UT1000 Project dataset highlights the need for future work to explore the generalizability of FENAP across different domains and datasets. Future research should also consider integrating additional data sources such as physiological signals and environmental sensors to create a more comprehensive behavioral analysis platform. With its proven effectiveness, FENAP has the potential to drive innovative solutions and significantly improve our understanding of human behavior in natural settings.

## A Appendices

### A.1 User Prompt Example

```
You are provided with a JSON file containing survey
    data from a participant. Each key in the JSON file
    is briefly explained below:

local.date: Date of data collection.
loc.var: Square root of the variance in GPS
    coordinates, indicating movement variability.
...

gender, age: Demographic information.

{json}

Summarize the participant's day, focusing on their
    emotional well-being, physical activity, and social
    interactions.
```

**Code Listing 1** Narrative Construction Prompt. The `{json}` placeholder represents the actual JSON data to be inserted into the prompt.

### A.2 System Prompt Configuration

```
You are a helpful, respectful and honest assistant. Always answer as
    helpfully as possible, while being safe. Your answers should not
    include any harmful, unethical, racist, sexist, toxic, dangerous,
    or illegal content. Please ensure that your responses are
    socially unbiased and positive in nature.
```

**Code Listing 2** System prompt used in LLaMA-2-70b-chat

### A.3 Example JSON Structures

```
{
"local.date": "2018-10-16",
"day_of_week": "Tuesday",
"loc.var": 0.003215511785567,
"ent.pls": 0.127980255944312,
"content_x": 2,
"energy_x": 2,
"lonely_x": 0,
"refreshed": 2,
"restful": 3,
"sad_x": 0,
"stress_x": 0,
"where.at": "[Home (dorm; apartment)]",
"with.whom": "[Roommates]",
"doing.what": "[Browsing the Internet; using social
   media; Eating; drinking]",
"interact.how": "[Talking in person; Interacting on
   Instagram]",
"gender": "Male",
"age": 18
}
\label{lis:promptConstruction}
```

**Code Listing 3** Example JSON for FENAP

```
{
"day_of_week": "4",
"loc.var": 0.0293826174881786,
"ent.pls": 0.0999830421533941,
"missing.hours": 3,
"content_x": 3,
"energy_x": 3,
"lonely_x": 2,
"refreshed": 2,
"restful": 3,
"sad_x": 1,
"stress_x": 1,
"where.at.Gym": 0,
"where.at.Store___Mall": 0,
"where.at.Work": 0,
...
"where.at.NO_ANSWER_SELECTED": 0,
"where.at.Campus": 0,
...
"with.whom.alone": 0,
"with.whom.1": 0,
...
"with.whom.NO_ANSWER_SELECTED": 0,
"doing.what.napping": 0,
...
"doing.what.NO_ANSWER_SELECTED": 0,
"interact.how.alone": 1,
...
"interact.how.NO_ANSWER_SELECTED": 0
"gender": 1,
"age": 20
}
```


**Code Listing 4** Example JSON for Baseline

**Author Contributions** F.Z. (Fengxiang Zhao) conducted the majority of the research activities, including the design and execution of experiments, data acquisition, analysis, and interpretation, and drafting the manuscript. Y.S. (Yi Shang) and T.J.T. (Timothy J. Trull) contributed to reviewing the work, providing critical feedback, guidance, and expertise that shaped the research, analysis, and manuscript.

**Funding** The work has been supported by NIH grants R01 DA055654 and R01 AA029094.

**Data Availability** The datasets analyzed during the current study are publicly available and have been cited appropriately within the manuscript. Full references to the datasets are provided in the bibliography section to ensure transparency and reproducibility.

## Declarations

**Ethical Approval** Not applicable.

**Conflict of Interest** The authors declare no competing interests.

## References


1. Shiffman S, Stone A, Hufford M (2008) Ecolocial momentary assessment. Annu Rev Clin Psychol 4:1–32. https://doi.org/10.1146/annurev.clinpsy.3.022806.091415
2. Wu C, Fritz H, Nagy Z, Maestre JP, Thomaz E, Julien C, Castelli DM, de Barbaro K, Harari GM, Craddock RC, Kinney KA, Gosling SD, Schnyer DM (2020) Multi-modal data collection for measuring health, behavior, and living environment of large-scale participant cohorts: conceptual framework and findings from deployments. arXiv
3. Hastie T, Tibshirani R, Friedman J (2009) The elements of statistical learning: data mining, inference and prediction, 2nd edn. Springer, New York. http://www-stat.stanford.edu/~tibs/ElemStatLearn/
4. Richardson B, Fuller-Tyszkiewicz M, O'Donnell R, Ling M, Staiger P (2017) Regression tree analysis of ecological momentary assessment data. Health Psychol Rev 11:235–241. https://doi.org/10.1080/17437199.2017.1343677
5. Hart A, Reis D, Prestele E, Jacobson N (2022) Using smartphone sensor paradata and personalized machine learning models to infer participants' well-being: ecological momentary assessment. J Med Internet Res 24. https://doi.org/10.2196/34015
6. Kim H, Lee S, Lee S, Hong S, Kang H, Kim N (2019) Depression prediction by using ecological momentary assessment, actiwatch data, and machine learning: observational study on older adults living alone. JMIR mHealth and uHealth 7. https://doi.org/10.2196/14149
7. Narkhede SM, Luther L, Raugh IM, Knippenberg AR, Esfahlani FZ, Sayama H, Cohen AS, Kirkpatrick B, Strauss GP (2021) Machine learning identifies digital phenotyping measures most relevant to negative symptoms in psychotic disorders: implications for clinical trials. Schizophr Bull 48(2):425–436. https://doi.org/10.1093/schbul/sbab134 https://academic.oup.com/schizophreniabulletin/article-pdf/48/2/425/42645893/sbab134.pdf
8. Spanakis G, Weiss G, Roefs A (2016) Bagged boosted trees for classification of ecological momentary assessment data. ArXiv abs/1607.01582. https://doi.org/10.3233/978-1-61499-672-9-1612
9. Aminikhanghahi S, Schmitter-Edgecombe M, Cook D (2020) Context-aware delivery of ecological momentary assessment. IEEE J Biomed Health Inform 24:1206–1214. https://doi.org/10.1109/JBHI.2019.2937116
10. Qiu X, Sun T, Xu Y, Shao Y, Dai N, Huang X (2020) Pre-trained models for natural language processing: a survey. SCIENCE CHINA Technol Sci 63:1872–1897. https://doi.org/10.1007/s11431-020-1647-3
11. Zhou C, Li Q, Li C, Yu J, Liu Y, Wang G, Zhang K, Ji C, Yan Q, He L, Peng H, Li J, Wu J, Liu Z, Xie P, Xiong C, Pei J, Yu PS, Sun L (2023) A comprehensive survey on pretrained foundation models: a history from BERT to ChatGPT
12. Narayan A, Chami I, Orr LJ, R'e C (2022) Can foundation models wrangle your data? Proc VLDB Endow 16:738–746. https://doi.org/10.48550/arXiv.2205.09911
13. Sui Y, Zhou M, Zhou M, Han S, Zhang D (2024) Table meets LLM: can large language models understand structured table data? A Benchmark and Empirical Study
14. Cloutier NA, Japkowicz N (2023) Fine-tuned generative LLM oversampling can improve performance over traditional techniques on multiclass imbalanced text classification. In: 2023 IEEE International conference on big data (BigData), pp 5181–5186. https://doi.org/10.1109/BigData59044.2023.10386772
15. Fang X, Xu W, Tan FA, Zhang J, Hu Z, Qi Y, Nickleach S, Socolinsky D, Sengamedu S, Faloutsos C (2024) Large Language Models(LLMs) on tabular data: prediction, generation, and understanding – a survey
16. McMaster C, Liew DF, Pires DE (2023) Adapting pretrained language models for solving tabular prediction problems in the electronic health record

17. Devlin J, Chang M-W, Lee K, Toutanova K (2018) BERT: pre-training of deep bidirectional transformers for language understanding
18. Liu Y, Ott M, Goyal N, Du J, Joshi M, Chen D, Levy O, Lewis M, Zettlemoyer L, Stoyanov V (2019) RoBERTa: a robustly optimized BERT pretraining approach
19. Chung HW, Hou L, Longpre S, Zoph B, Tay Y, Fedus W, Li Y, Wang X, Dehghani M, Brahma S, Webson A, Gu SS, Dai Z, Suzgun M, Chen X, Chowdhery A, Castro-Ros A, Pellat M, Robinson K, Valter D, Narang S, Mishra G, Yu A, Zhao V, Huang Y, Dai A, Yu H, Petrov S, Chi EH, Dean J, Devlin J, Roberts A, Zhou D, Le QV, Wei J (2022) Scaling instruction-finetuned language models
20. Touvron H et al (2023) Llama 2: open foundation and fine-tuned chat models. arXiv:2307.09288
21. Wu C (2020) Anonymous daily features calculated from GPS data (Beiwe). https://doi.org/10.18738/T8/1TDP7I
22. Wu C (2020) Ecological momentary assessment data (Beiwe). https://doi.org/10.18738/T8/OPQMF3
23. Wu C (2020) Accelerometer data (Beiwe). https://doi.org/10.18738/T8/F9FHZS
24. Wu C (2020) Participants key file. https://doi.org/10.18738/T8/OUUPIA
25. Gu Y, Tinn R, Cheng H, Lucas M, Usuyama N, Liu X, Naumann T, Gao J, Poon H (2020) Domain-specific language model pretraining for biomedical natural language processing. arXiv preprint arXiv:2007.15779
26. Lee J, Yoon W, Kim S, Kim D, Kim S, So CH, Kang J (2020) Biobert: a pre-trained biomedical language representation model for biomedical text mining. Bioinformatics 36(4):1234–1240
27. Raffel C, Shazeer N, Roberts A, Lee K, Narang S, Matena M, Zhou Y, Li W, Liu PJ (2023) Exploring the limits of transfer learning with a unified text-to-text transformer
28. Hu EJ, shen, Wallis P, Allen-Zhu Z, Li Y, Wang S, Wang L, Chen W (2022) LoRA: low-rank adaptation of large language models. In: International conference on learning representations. https://openreview.net/forum?id=nZeVKeeFYf9
29. Quinlan JR (1986) Induction of decision trees. Mach Learn 1(1):81–106. https://doi.org/10.1007/BF00116251
30. Breiman L (2001) Random forests. Mach Learn 45:5–32. https://doi.org/10.1023/A:1010950718922
31. Cortes C, Vapnik VN (1995) Support-vector networks. Mach Learn 20:273–297
32. Freund Y, Schapire RE (1997) A decision-theoretic generalization of on-line learning and an application to boosting. J Comput Syst Sci 55(1):119–139. https://doi.org/10.1006/jcss.1997.1504
33. Chen T, Guestrin C (2016) Xgboost: a scalable tree boosting system. In: Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining. KDD '16. ACM, New York, NY, USA pp 785–794. https://doi.org/10.1145/2939672.2939785
34. Ke G, Meng Q, Finley T, Wang T, Chen W, Ma W, Ye Q, Liu T-Y (2017) Lightgbm: a highly efficient gradient boosting decision tree. In: Neural information processing systems. https://api.semanticscholar.org/CorpusID:3815895
35. Ostroumova L, Gusev G, Vorobev A, Dorogush AV, Gulin A (2017) Catboost: unbiased boosting with categorical features. In: Neural information processing systems. https://api.semanticscholar.org/CorpusID:5044218
36. Rumelhart DE, Hinton GE, Williams RJ (1986) Learning representations by back-propagating errors. Nature 323(6088):533–536. https://doi.org/10.1038/323533a0
37. Guo H, Tang R, Ye Y, Li Z, He X (2017) Deepfm: a factorization-machine based neural network for ctr prediction. ArXiv abs/1703.04247
38. Arik SÖ, Pfister T (2019) Tabnet: attentive interpretable tabular learning. ArXiv abs/1908.07442
39. Mangrulkar S, Gugger S, Debut L, Belkada Y, Paul S, Bossan B (2022) PEFT: state-of-the-art parameter-efficient fine-tuning methods. https://github.com/huggingface/peft
40. Zheng Y, Zhang R, Zhang J, Ye Y, Luo Z, Ma Y (2024) Llamafactory: unified efficient fine-tuning of 100+ language models. arXiv preprint arXiv:2403.13372