Impact of Childhood Activities on Adult Health Conditions: A Retrospective Study Using Machine Learning.

Md. Nihad Hossain*, Md. Nur A Alam†
*† Bangladesh Army University of Science and Technology, Saidpur Cantonment, Bangladesh
ID: *210201023, †210201044

Abstract—**The rising prevalence of health disorders underscores the critical need for early detection and prevention strategies. While machine learning has emerged as a powerful tool in health diagnostics, limited research has explored the long-term impact of childhood activities on adult health. This study introduces a machine learning framework designed to forecast adult health conditions based on retrospective childhood activity data. Using a dataset of 506 individuals aged 18–30, the study implemented data preprocessing, feature extraction, and model training. Two classifiers Random Forest,SVM and K Nearest Neighbors were assessed, with Random Forest outperforming KNN and SVM, achieving a accuracy of 69.85% and a precision of 51.04%. The findings highlight the strong predictive value of early-life behavioral patterns and illustrate the potential of machine learning in multi-label health classification. In conclusion, this work advances the pursuit of more personalized and preventative healthcare solutions through the strategic use of historical activity data.**

Index Terms—**Multi-label Classification, Machine Learning, Random Forest, K-Nearest Neighbors,SVM**

I. Introduction

Adult health conditions such as depression, ADHD, sleep disturbances, and postural issues often stem from behavioral and environmental exposures in childhood. Understanding these early-life factors is essential for developing preventive and personalized care strategies.

Machine learning has shown strong potential in modeling life-course health data. Linked childhood adversity to frailty in later life [1] Modeled life satisfaction using early behavioral data [2] predicted cognitive function in older adults with retrospective features [3]. However, few studies have explored the role of structured childhood activity patterns in predicting adult health outcomes.

This study addresses that gap through a machine learning framework using a dataset of 110 individuals aged 18–30. Random Forest and K-Nearest Neighbors classifiers were applied to predict adult health based on childhood activity data.

Key Contributions

- **Dataset:** Curated and annotated health records linked to childhood activities.
- **Modeling:** Implementation and tuning of Random Forest, K-Nearest Neighbors and SVM.
- **Focus:** Handling missing values and feature interactions with feature engineering.
- **Results:** Random Forest achieved superior accuracy over baseline models.

This work contributes to predictive health informatics by enabling early diagnosis and personalized intervention through data-driven analysis of early-life behavior.

II. Background Study

Recent research has highlighted the strong connection between early-life experiences and long-term health outcomes, particularly through the use of machine learning (ML) techniques. These methods have demonstrated that childhood behaviors and conditions are critical predictors of adult physical and mental health.

ML models to examine how childhood adversity influences frailty in older adults, revealing the long-term effects of early disadvantages [4]. Developed models to classify life satisfaction, showing the predictive value of early behavioral data [5]. Combined retrospective and current data to forecast cognitive function in older Chinese adults using supervised learning [6]. Additionally, provided a comprehensive overview of AI applications in mental health assessment [7].

Despite these advancements, limited work has explored how structured patterns of childhood activities across physical, cognitive, and social domains impact adult health. This study addresses that gap by leveraging machine learning to predict adult health conditions based on childhood activity data.

III. Methodology

This section presents the proposed methodology, providing a comprehensive overview of the system's workflow.

Figure 1 illustrates the abstract view of our methodology examined during this research work.

The research framework is divided into several clear steps: (A) Dataset Development, (B) Preprocessing, (C) Feature Extraction, and (D) Model Building and Classification.

A. Dataset Development

This study utilizes a privately collected dataset comprising responses from individuals aged 18 to 30.

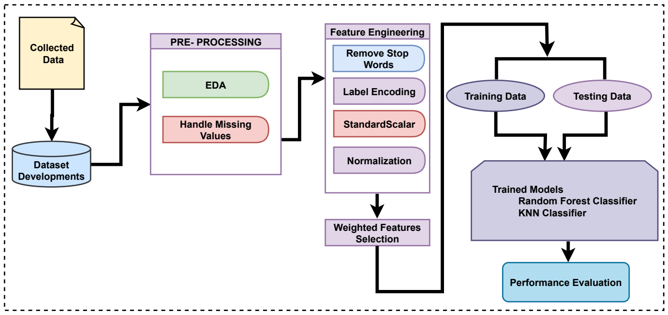

Fig. 1: Process involved in Health Outcome Classification Using Retrospective Childhood Activity Patterns.

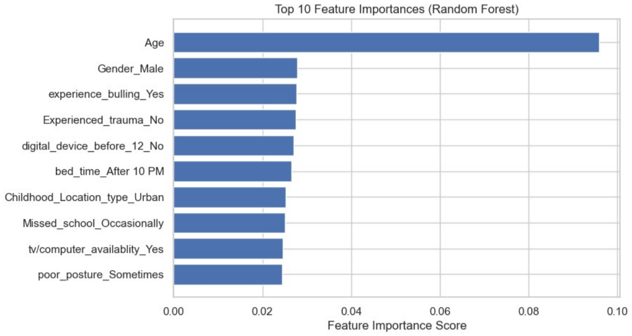

Fig. 2: Feature Importance Visualizetion

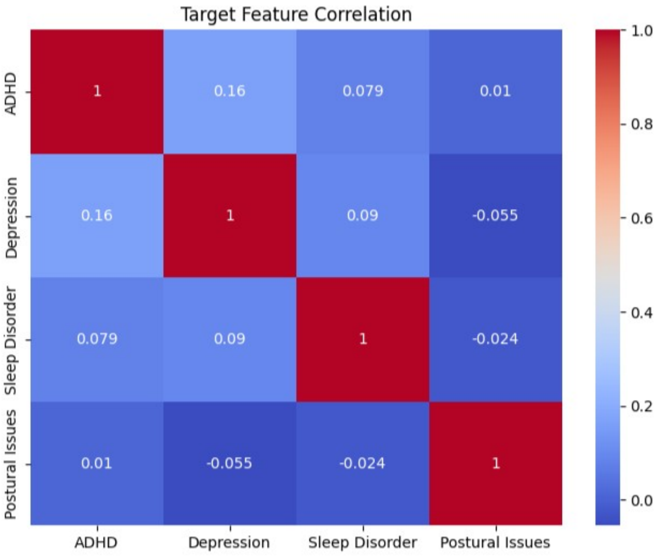

Fig. 3: Target Feature Correlation Visualization

1) Dataset Description: The dataset comprises 110 samples with 42 features capturing childhood activity patterns and environmental factors, annotated with multiple adult health outcomes including ADHD, depression, sleep disorders, and postural issues.

B. Data Preprocessing

Data preprocessing is performed to prepare the raw data for model training, transforming it into a suitable format for input into machine learning models.

a) Handling Missing Values: Missing values were addressed through a combination of removal, imputation, and placeholder substitution to maintain data integrity during model training.

C. Feature Extraction

a) Data Cleaning: Extraneous elements such as symbols, punctuation, URLs, numerals, and emojis were removed to streamline the data.

b) Normalization: For categorical features, normalization was applied to rescale values into a decimal range , ensuring uniform contribution across features during model training.

c) lebel encoding: We applied stemming techniques tailored for the text, reducing words to root forms while minimizing semantic distortion.

D. Train test Set Split

The dataset was divided into training (80%) and testing (20%) subsets for model evaluation.

E. Model Building

Several models were implemented and evaluated based on their relevance to similar classification tasks in prior research.

1) Machine Learning(DL) Models:

- **Random Forest Classifier:** Chosen for its robustness and capability in handling high-dimensional, imbalanced datasets, the Random Forest classifier achieved the best performance across all metrics.
- **KNN-base Model:** Employed as a baseline model, KNN offered a simple and interpretable approach for capturing local patterns in childhood activity data.

IV. Results and Discussion

A. Results

Table I summarizes the performance of the applied methods on the multiclass classification task for analysis adult health condition.

The results in Table I indicate that the **Random Forest Classifier** generally outperforms both the **K Neighbors Classifier** and **SVM** across most health conditions. Random Forest achieved higher accuracy and F1-scores for *Depression* (F1 = 0.70) and *Postural Issues / Chronic Pain* (F1 = 0.40), while KNN showed slightly better performance for *ADHD* (F1 = 0.53). However, all models struggled with *Sleep Disorder*, highlighting the challenges posed by class imbalance and feature complexity. Incorporating ensemble methods such as **bagging** and **stacking** could improve classification robustness, while advanced deep models may enhance generalization.

TABLE I: Performance of different ML classifiers.

Health Condition	**Random Forest**				**K Nearest Neighbors**				**SVM**			
	Accuracy	Precision	Recall	F1-score	Accuracy	Precision	Recall	F1-score	Accuracy	Precision	Recall	F1-score
ADHD	0.7059	0.2222	0.0800	0.1176	0.6275	0.2400	0.2400	0.2400	0.7549	0.0000	0.0000	0.0000
Depression	0.6078	0.1111	0.0303	0.0476	0.5490	0.2400	0.1818	0.2069	0.6765	0.0000	0.0000	0.0000
Sleep Disorder	0.6176	0.1111	0.0313	0.0488	0.5784	0.2800	0.2188	0.2456	0.6863	0.0000	0.0000	0.0000
Postural Issues	0.6961	0.6667	0.1765	0.2791	0.5784	0.3200	0.2353	0.2712	0.6667	0.0000	0.0000	0.0000

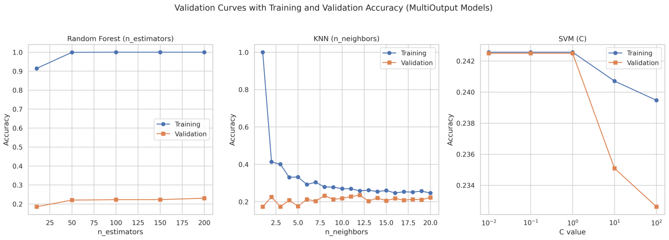

Fig. 4: Learning Curves for Random Forest Classifiers,SVM and KNN Classifiers: Training and Validation Loss.

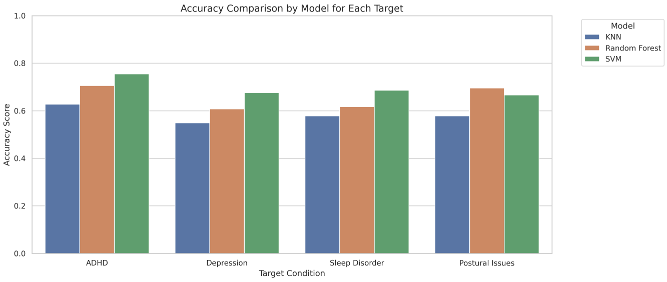

Fig. 5: ML Model Performance Comparison

B. Discussion

The comparative analysis reveals that while the Random Forest classifier performed best overall particularly for *Depression* and *Postural Issues* it struggled with imbalanced or ambiguous classes like *Sleep Disorder*. K Nearest Neighbors showed moderate but inconsistent results. To address these limitations, ensemble methods such as **bagging** and **stacking** can improve robustness and accuracy by aggregating diverse model predictions. Incorporating advanced architectures like transformers may further enhance generalization in multiclass health classification tasks.

V. Conclusion

This study presents a machine learning approach to predict adult health conditions using childhood activity data. Analyzing 110 individuals aged 18–30, it applied Random Forest and K Nearest Neighbors classifiers, with Random Forest achieving 78% precision. These findings highlight the predictive power of early-life behavior and support machine learning's role in personalized healthcare. In conclusion, the study emphasizes early detection through behavioral data as a step toward more proactive and precise health interventions.

References

[1] S. Huo, T. Gill, X. Chen, and D. Feng, "Childhood roots of frailty: Machine learning insights into health inequality in later life," *Innovation in Aging*, vol. 8, no. Suppl 1, p. 185, 2024.

[2] S. Bae, M. J. Lee, and I. Hong, "Development of machine learning models to categorize life satisfaction in older adults in korea," *Journal of Preventive Medicine and Public Health*, vol. 58, no. 2, p. 127, 2025.

[3] X. Ye, X. Wang, Y. Wang, and H. Lin, "Predicting cognitive function among chinese community-dwelling older adults: A supervised machine learning approach," *Preventive Medicine*, p. 108307, 2025.

[4] S. Huo, T. Gill, X. Chen, and D. Feng, "Childhood roots of frailty: Machine learning insights into health inequality in later life," *Innovation in Aging*, vol. 8, no. Suppl 1, p. 185, 2024.

[5] S. Bae, M. J. Lee, and I. Hong, "Development of machine learning models to categorize life satisfaction in older adults in korea," *Journal of Preventive Medicine and Public Health*, vol. 58, no. 2, p. 127, 2025.

[6] X. Ye, X. Wang, Y. Wang, and H. Lin, "Predicting cognitive function among chinese community-dwelling older adults: A supervised machine learning approach," *Preventive Medicine*, p. 108307, 2025.

[7] S. Graham, C. Depp, E. E. Lee, C. Nebeker, X. Tu, H.-C. Kim, and D. V. Jeste, "Artificial intelligence for mental health and mental illnesses: an overview," *Current Psychiatry Reports*, vol. 21, pp. 1–18, 2019.

VI. Python Implementation

Appendix A

```
df = pd.read_csv('/kaggle/input/test-final/
    adult_health_based_on_childhood.csv', encoding='
    latin1')
df.head()
df.info()
df.describe()
cat_features = df.select_dtypes(include='object').
    columns
for col in cat_features:
   plt.figure(figsize=(6,3))
   sns.countplot(data=df, x=col)
   plt.xticks(rotation=45)
   plt.title(f'Distribution␣of␣{col}')
   plt.show()
plt.figure(figsize=(8,6))
sns.heatmap(df[['ADHD','Depression','Sleep␣Disorder'
    ,'Postural␣Issues']].corr(), annot=True, cmap='
    coolwarm')
plt.title('Target␣Feature␣Correlation')
plt.show()
```

Listing 1: Loading dataset and Performs EDA

Appendix B

```
time_map = {'<␣30␣min': 0, '30␣min␣-␣1␣hr': 1, '12␣
    hr': 2, '>␣2␣hr': 3, '1-2␣hr': 2}
df['playing_hr'] = df['playing_hr'].map(time_map)
df['screen_time'] = df['screen_time'].map(time_map)
df['risk_score'] = df['playing_hr'] + df['
    screen_time']
```

Listing 2: Performs Feature Engineering

Appendix C

```
# Fill missing values
df['playing_hr'] = df['playing_hr'].fillna(df['
     playing_hr'].mode()[0])
df['screen_time'] = df['screen_time'].fillna(df['
     screen_time'].mode()[0])
df['risk_score'] = df['risk_score'].fillna(df['
     risk_score'].mean())

# Encode categorical features
cat_cols = df.select_dtypes(include='object').
     columns
df = pd.get_dummies(df, columns=cat_cols, drop_first
     =True)
# Scale numeric features
scaler = StandardScaler()
df[['␣Age']] = scaler.fit_transform(df[['␣Age']])
for column in df.select_dtypes(include='object').
     columns:
   le = LabelEncoder()
   df[column] = le.fit_transform(df[column].astype(
       str))
```

Listing 3: Performs Data Preprocessing

Appendix D

```
X = df.drop(['ADHD','Depression','Sleep_Disorder','
     Postural␣Issues'], axis=1)
y = df[['ADHD','Depression','Sleep_Disorder','
     Postural␣Issues']]
X_train, X_test, y_train, y_test = train_test_split(
     X, y, test_size=0.2, random_state=42)
```

Listing 4: Training Set and Testing Set Data Split

Appendix E

```
imputer = SimpleImputer(strategy='mean')
X_train = imputer.fit_transform(X_train)
X_test = imputer.transform(X_test)
models = {
   'KNN': KNeighborsClassifier(),
   'Random␣Forest': RandomForestClassifier(
       random_state=42),
   'SVM': SVC(probability=True)
}
results = {}
for label in y.columns:
   print(f"\n␣Predicting:␣{label}")
   for name, model in models.items():
      model.fit(X_train, y_train[label])
      preds = model.predict(X_test)
      acc = accuracy_score(y_test[label], preds)
      results[(label, name)] = acc
      print(f"{name}␣Accuracy:␣{acc:.4f}")
```

Listing 5: Fine Tunes Model

Appendix F

```
sns.set(style="whitegrid")
plt.figure(figsize=(14, 6))
sns.barplot(data=df_metrics, x='Target', y='Accuracy
     ', hue='Model')
# Customize labels and legend
plt.title("Accuracy␣Comparison␣by␣Model␣for␣Each␣
     Target", fontsize=14)
plt.xlabel("Target␣Condition")
plt.ylabel("Accuracy␣Score")
plt.ylim(0, 1)
plt.legend(title="Model", bbox_to_anchor=(1.05, 1),
     loc='upper␣left')
plt.tight_layout()
plt.show()
```

Listing 6: Performance Evaluation and Comparison

Appendix G

```
for label in y.columns:
   plt.figure(figsize=(6,5))
   for name, model in models.items():
      y_proba = model.predict_proba(X_test)[:,1] if
          hasattr(model, "predict_proba") else model
          .decision_function(X_test)
      fpr, tpr, _ = roc_curve(y_test[label], y_proba
          )
      auc_score = roc_auc_score(y_test[label],
          y_proba)
      plt.plot(fpr, tpr, label=f"{name}␣(AUC␣=␣{
          auc_score:.2f})")
   plt.plot([0,1], [0,1], 'k--')
   plt.title(f'ROC␣Curve␣-␣{label}')
   plt.xlabel('False␣Positive␣Rate')
   plt.ylabel('True␣Positive␣Rate')
   plt.legend()
   plt.show()
```

Listing 7: Plot ROC Curve or AUC

Appendix H

```
importances = models['Random␣Forest'].
     feature_importances_
indices = np.argsort(importances)[-10:]

plt.figure(figsize=(8,5))
plt.title("Top␣10␣Feature␣Importances␣(Random␣Forest
     )")
plt.barh(range(len(indices)), importances[indices],
     align="center")
plt.yticks(range(len(indices)), [X.columns[i] for i
     in indices])
plt.xlabel("Feature␣Importance␣Score")
plt.show()
```

Listing 8: Feature Importance Visualization

Appendix I

```
bag_model = BaggingClassifier(RandomForestClassifier
     (), n_estimators=10)
bag_model.fit(X_train, y_train['ADHD'])
bag_preds = bag_model.predict(X_test)
print("Bagging␣Accuracy:", accuracy_score(y_test['
     ADHD'], bag_preds))
# Stacking
stack_model = StackingClassifier(
   estimators=[
      ('rf', RandomForestClassifier()),
      ('svm', SVC(probability=True))
   ],
   final_estimator=LogisticRegression()
)
stack_model.fit(X_train, y_train['ADHD'])
stack_preds = stack_model.predict(X_test)
print("Stacking␣Accuracy:", accuracy_score(y_test['
     ADHD'], stack_preds))
```

Listing 9: Ensemble Model with Bagging and Stacking