Predicting Physical Fatigue in Hajj Pilgrims Using Wearable Sensors and Neural Network

Faris Safar R. Alkhudaydi
College of Computer and Information Science, Majmaah University
Majmaah, KSA
471100160@s.mu.edu.sa

Emad Abaalkhail*, Hussain Alali, Fawaz Alotaibi, Sultan Alzahrani*
Digital Health Institute, King Abdulaziz City for Science and Technology
Riyadh, KSA
{ealamoud, halali, falotaibe, szahrani}@kacst.gov.sa
Majed Alghamdi
General Diroctrate of National Security, King Abdulaziz City for Science and Technology
Riyadh, KSA
msghamdi@kacst.gov.sa

***Abstract*—Every year, over 2 million Muslims travel to Mecca for Hajj. Walking and Performing ritual duties (Manasek like Kaaba circuling, Walk between Safaa and Marwa, Stoning, etc)for hours in extreme heat (often 45°C) causes serious health risks. Physical fatigue is usually the first warning sign before someone collapses. In this work, we replicated a 2024 study by Al-Shaery et al. that used wearable sensors and neural networks to predict fatigue levels in pilgrims and found a major problem as we were experimenting: the dataset was sorted by fatigue level. When we initially trained without fixing this issue, we got 99%+ accuracy, which was higher than Al-Shaery et al.'s (reported 85.27%). This seemed suspicious at first. After investigation, we discovered the sorted data created fake patterns where the model learned row position instead of physiological signal as we deal with Neural Network and training through batching where data fed to the model for training batch by batch. Once we shuffled the data properly and used only 7 raw sensor measurements (avoiding engineered features that might leak information), our accuracy dropped to a realistic 88.15% on 100,000 test samples, still overcoming the original paper's 85.27% by 2.88 percentage points, but for the right reasons.**

The model could help monitor pilgrims in real-time and flag those at risk before medical emergencies happen. However, it still misses 22% of severe fatigue cases, which is concerning for safety. We discuss these problems and suggest improvements.

I. Introduction

A. The Problem

Picture millions of people, many elderly, walking for hours in a about 45°C heat. That's Hajj. Heat exhaustion, dehydration, and heart problems are constant risks. Medical staff can't manually watch millions of people at once.

Wearable sensors might help. Devices like the Empatica E4 wristband track heart rate, skin moisture, temperature, and movement continuously. Combined with machine learning, they could predict who's about to collapse before it happens. The challenge is building models that actually work in the real world.

B. Previous Work

Al-Shaery et al. (2024) collected 16 million measurements from Hajj pilgrims and trained a neural network to classify fatigue into three levels: not tired, moderately tired, and very tired [1]. They reported 85.27% accuracy. When we first attempted replication without careful data handling, we got 99%+ accuracy—suspiciously high. This led us to investigate potential issues: Were they using engineered features calculated from the target variable itself? Was the data properly shuffled? Our investigation revealed that data sorting was the culprit behind inflated accuracy.

Other studies have similar problems. Some use features calculated from what they're trying to predict (circular logic). Others don't account for time ordering in data [2]. These mistakes mean models that work in labs but fail when deployed.

C. Our Approach

We took a systematic approach to replication:

1) **Understand the original:** We studied Al-Shaery et al.'s methodology to identify what worked and what might be improved.
2) **Investigate data quality:** We examined the dataset structure for potential issues like sorting or leakage.
3) **Simplify features:** We tested whether raw sensor data alone could outperform engineered features.
4) **Apply best practices:** We used proper shuffling, stratification, and class weighting.
5) **Validate rigorously:** We evaluated on held-out test data with per-class metrics.

D. Key Contributions

- Identified and fixed a critical data sorting issue that created fake 99%+ accuracy (vs. Al-Shaery's realistic 85.27%)

- Demonstrated that 7 raw sensors outperform 18+ engineered features when data leakage is controlled
- Achieved 88.15% accuracy with proper methodology, surpassing the original paper's 85.27% by 2.88 points
- Provided detailed analysis of failure modes, especially the 22% miss rate for severe fatigue
- Offered practical recommendations for real-world deployment

II. The Dataset

A. Dataset Selection Process

We evaluated multiple Hajj-related datasets before selecting one. Each dataset served different purposes:

Hajj healthQA [3]: Contains 341,000+ multilingual medical Q&A pairs. Excellent for building chatbots or question-answering systems, but lacks the continuous physiological measurements needed for fatigue prediction.

Hajj Crowd Management [4]: Includes crowd behavior, environmental conditions, and aggregate health metrics. Designed for macro-level crowd prediction rather than individual-level health monitoring.

Hajj Crowd Density [5]: Provides 27,000 annotated crowd images. Useful for computer vision applications but doesn't contain wearable sensor data.

Epidemiological datasets [6]–[8]: Offer population-level statistics on disease incidence and mortality. Valuable for public health policy but not suitable for real-time individual prediction.

We selected the Hajj Crowd Activity Prediction Dataset [9] because it uniquely combines:

1) Multimodal physiological signals from research-grade wearables
2) High temporal resolution (up to 64 Hz sampling)
3) Sufficient scale (16+ million samples from 64 participants)
4) Pre-labeled fatigue levels (enabling supervised learning)
5) Public accessibility (no lengthy approval process)
6) Prior validation (benchmark from published work)

B. Data Characteristics

The dataset originates from the 2023 Hajj season. Researchers equipped 64 volunteer pilgrims with Empatica E4 wristbands during simulated Hajj rituals. These medical-grade devices recorded 32 different features at 1 Hz over multiple days, yielding 16,134,883 total samples.

C. Feature Categories

Raw Physiological Sensors (7 features):
These represent direct hardware measurements:

- `bvp`: Blood volume pulse from photoplethysmography sensor (range: -0.5 to 0.5)
- `gsr_x`: Galvanic skin response measuring sweat gland activity (range: 0.01 to 20 μS)
- `temp`: Skin temperature at wrist (range: 28–38°C)
- `x`, `y`, `z`: Three-axis accelerometer data (range: -2g to +2g per axis)
- `ibi`: Inter-beat interval between heartbeats (range: 400–1200 ms)

Engineered Features (11 total):
Table I lists all engineered features. We excluded these from our model because they may incorporate fatigue information in their calculations, creating circular reasoning.

TABLE I. Engineered Features and Why We Excluded Them

Feature Name	Why Excluded
`heartRate`	Redundant with raw IBI
`respiration-Rate`	Redundant with raw BVP
`hrv`	Redundant with raw IBI
`skinTemp`	Redundant with raw temp
`bloodVolume-Pulse`	Redundant with raw BVP
`stressIndex`	May use target variable
`activityLevel`	May use target variable
`energy-Expenditure`	May use target variable
`stepCount`	Too predictive (leakage)
`distance`	Derived from stepCount
`calories`	Same as energyExpenditure

Contextual Features (14 features):
Location, environmental, and temporal data. We also excluded these to focus on physiological signals only.

D. Target Variable

The `physicalTiredLevel` variable has three classes:

- **Level 1 (Not Tired):** 41.52% of samples
- **Level 2 (Moderately Tired):** 9.73% of samples (minority class)
- **Level 3 (Very Tired):** 48.75% of samples

E. Critical Data Issue: The 99% vs 85.27% Mystery

When we first examined the data, we discovered it was sorted by target variable. All Level 1 samples appeared first, followed by Level 2, then Level 3. This creates severe data leakage: a model can achieve 99%+ accuracy simply by learning row position rather than physiological patterns, plus we trained on sample of data where classes are severely imbalanced.

We verified this empirically:

- **Without shuffling (flawed):** 99.2% accuracy
- **Al-Shaery et al. (2024):** 85.27% accuracy (proper methodology)
- **Our corrected approach from our previous approach:** 88.15% accuracy (with shuffling)

The 11-point drop from 99.2% to 88.15% reveals the magnitude of the leakage. Our 88.15% beats Al-Shaery's 85.27% because we used only raw sensors (avoiding potential leakage from engineered features) and applied rigorous class weighting.

F. Data Preparation

Our preprocessing pipeline:

1) **Shuffling:** Randomized sample order with fixed seed (42) for reproducibility

2) **Sampling:** Selected 500,000 rows (computational constraints of free Google Colab)
3) **Feature selection:** Retained only 7 raw sensors
4) **Missing values:** Applied forward-fill (last observation carried forward)
5) **Train/test split:** 80/20 split with stratification to maintain class balance
6) **Standardization:** Z-score normalization using training set statistics only

III. Methodology

A. Model Architecture

We designed a simple feedforward neural network:

- **Input layer:** 7 features (raw sensors only)
- **Hidden layer 1:** 64 neurons with ReLU activation, L2 regularization (0.01), batch normalization, 30% dropout
- **Hidden layer 2:** 32 neurons with ReLU activation, L2 regularization (0.01), batch normalization, 30% dropout
- **Output layer:** 3 neurons with softmax activation (one per fatigue level)

Total parameters: 2,659. This compact architecture can run on mobile devices.

B. Training Configuration

- **Loss function:** Categorical cross-entropy
- **Optimizer:** Adam with learning rate 0.001
- **Batch size:** 128 samples
- **Class weights:** Level 1: 0.80×, Level 2: 3.43×, Level 3: 0.68× (computed from class frequencies)
- **Callbacks:** Early stopping (patience 10 epochs), learning rate reduction on plateau
- **Maximum epochs:** 100 (training stopped at epoch 24)

C. Experimental Setup

- **Hardware:** Google Colab with Tesla T4 GPU (free tier)
- **Data split:** 320,000 training, 80,000 validation, 100,000 test
- **Reproducibility:** All random operations seeded with 42
- **Training time:** 12 minutes 34 seconds
- **Inference time:** 0.08 seconds per 1,000 samples

The proposed model was developed to optimize the trade-off between prediction accuracy and computational efficiency, thereby making it carrier for deployment in environments with limited resources. We used a small feedforward neural network with ReLU activations and further employed batch normalization and dropout mechanisms to help the model generalize better and training becoming more stable.

The model is built upon only seven raw sensor features, which have been carefully chosen so that they represent the most important physiological signs that are related to the detection of fatigue but without overlapping and secondarily use of features which are derived and might introduce bias or circular reasoning. Thus the model is directly learning from the main measurements rather than from the indirectly coded ones. For training, categorical cross-entropy got used since it is good for multiple class classifications. As the optimizer, Adam (with learning rate = 0.001) was picked due to its reliable way of converging. Additionally, a batch size of 128 was selected as a good compromise between training stability and efficiency of computations. Class weights were added to the learning process in such a way that the classes which are rarely seen have definitely a higher impact on learning than the rest.

A mixture of interventions was implemented to minimize the possibility that the experimental results were due to chance. The data set was randomly reordered and partitioned to train, validate and test to avoid any ordering effect and to eliminate any chance of data leakage which was interpreted as an explanation of greater performance in the earlier models. Then stratified splitting was used to get the same class distribution in each of the train, validation, test sets. Also, early stopping and lowering of the learning rate were done in order to prevent overfitting and at the same time be able to support a more stable convergence. Besides that, random seeds were set to be able to reproduce the results.

All our experiments have been run on Google Colab and the used GPU was Tesla T4. Our final model is very small with only 2, 659 parameters and quick inference, so it can be used for real-time application smoothly.

IV. Results

A. Overall Performance

Table II shows our model's performance on the held-out test set of 100,000 samples.

TABLE II. Overall Model Performance on Test Set

Metric	Value
Test Accuracy	88.15%
Test Loss	0.2431
Training Accuracy	89.25%
Validation Accuracy	84.44%
Macro F1-Score	0.83
Weighted F1-Score	0.89

Our 88.15% accuracy surpasses Al-Shaery et al.'s 85.27% by 2.88 percentage points. The small gap between training (89.25%) and validation (84.44%) accuracy indicates minimal overfitting.

B. Accuracy Comparison: Understanding the Numbers

Table III clarifies the three different accuracy values readers will encounter in this paper.

Key insight: The 99.2% was not a success—it was a warning sign. Real-world accuracy should be in the 80–90% range for this problem. Anything higher suggests data leakage.

C. Model Comparison

We compared our neural network against four baseline machine learning models to validate our choice. All models used the same 7 raw sensor features, class weighting, and train/test split.

The neural network outperforms all baseline models. Random Forest came closest (86.42%), but still fell short by 1.73

TABLE III. ACCURACY COMPARISON: FLAWED VS PROPER METHODOLOGY

Approach	Details
Our initial (flawed)	99.2% accuracy; sorted data with no shuffling. Model learned row position, not physiology (fake result).
Al-Shaery et al.	85.27% accuracy reported in the original paper using proper methodology.
Our corrected	88.15% accuracy using shuffled data, raw sensors only, and class weighting (real result).
Improvement	+2.88% improvement over the reference paper due to simpler features and better handling of class imbalance.

TABLE IV. COMPARISON OF MACHINE LEARNING MODELS

Model	Performance Metrics
Neural Network	Accuracy: 88.15%, F1-Macro: 0.83, F1-Weighted: 0.89
Random Forest	Accuracy: 86.42%, F1-Macro: 0.79, F1-Weighted: 0.87
Gradient Boosting	Accuracy: 85.91%, F1-Macro: 0.78, F1-Weighted: 0.86
Logistic Regression	Accuracy: 82.34%, F1-Macro: 0.71, F1-Weighted: 0.83
SVM (RBF)	Accuracy: 81.67%, F1-Macro: 0.69, F1-Weighted: 0.82

percentage points. The simpler models (Logistic Regression and SVM) struggled with the non-linear relationships in physiological data.

Why neural networks won: They can learn complex interactions between sensors (for example, how heart rate variability relates to skin temperature under stress) that linear models miss. The class weighting and batch normalization also helped handle the severe imbalance.

D. Per-Class Analysis

Table V breaks down performance by fatigue level.

TABLE V. PER-CLASS PERFORMANCE METRICS

Class	Performance Metrics
Level 1	Precision: 1.00, Recall: 0.98, F1: 0.99, Support: 41,521
Level 2	Precision: 0.47, Recall: 0.99, F1: 0.63, Support: 9,730
Level 3	Precision: 0.98, Recall: 0.78, F1: 0.87, Support: 48,749

Level 1 (Not Tired): Near-perfect performance. High precision (1.00) and recall (0.98) indicate the model reliably identifies people who aren't fatigued.

Level 2 (Moderately Tired): Excellent recall (0.99) but low precision (0.47). The model catches almost all moderate fatigue cases but generates many false alarms. This reflects the challenge of the minority class (only 9.73% of data).

Level 3 (Very Tired): High precision (0.98) but concerning recall (0.78). The model misses 22% of severe fatigue cases. From a safety perspective, this is the most problematic failure mode.

E. Feature Importance Analysis

To understand which sensors contribute most to fatigue prediction, we analyzed feature importance using Random Forest's built-in importance scores (since neural networks don't provide direct feature importance).

Calculation method: We trained a Random Forest classifier (100 trees, max depth 20) on the same training data used for the neural network. Random Forest calculates feature importance using Mean Decrease in Impurity (MDI), also known as Gini importance. For each feature f and tree t:

$$Importance(f) = \frac{1}{T}\sum_{t=1}^{T}\sum_{n \in splits(f,t)} \Delta Gini(n) \quad (1)$$

where T is the number of trees (100), $splits(f,t)$ are all nodes in tree t that split on feature f, and $\Delta Gini(n)$ is the decrease in Gini impurity at node n. The final importance scores are normalized to sum to 100%.

Table VI shows the relative contribution of each raw sensor.

TABLE VI. FEATURE IMPORTANCE FOR FATIGUE PREDICTION

Feature	Details
ibi	Inter-beat interval, Importance: 28.3%
gsr_x	Skin conductance, Importance: 24.7%
temp	Skin temperature, Importance: 18.9%
bvp	Blood volume pulse, Importance: 12.4%
z	Vertical acceleration, Importance: 7.2%
x	Horizontal acceleration, Importance: 4.8%
y	Forward-backward acceleration, Importance: 3.7%

Key findings:

- **Heart rate variability (ibi)** is the strongest predictor (28.3%). Fatigue reduces heart rate variability—a well-established physiological marker.
- **Stress response (gsr_x)** ranks second (24.7%). Sweating increases with physical exertion and heat stress.
- **Temperature (temp)** contributes 18.9%. Core body temperature rises during prolonged activity.
- **Movement (x, y, z)** has lower importance (15.7% combined). While fatigue affects gait, the relationship is less direct than cardiovascular or thermoregulatory signals.

This ranking makes physiological sense: cardiovascular and autonomic nervous system markers (ibi, gsr_x) are more sensitive to fatigue than gross motor patterns.

F. Impact of Data Shuffling

Table VII demonstrates the critical importance of proper data handling.

The 11-point accuracy drop reveals that sorted data creates severe leakage. The 99.2% figure is meaningless—the model learns row position rather than physiological patterns.

TABLE VII. IMPACT OF DATA SHUFFLING ON MODEL ACCURACY

Condition	Accuracy
Without Shuffling	99.2%
With Proper Shuffling	88.15%
Difference	-11.05%

G. Training Dynamics

Training converged after 24 epochs (early stopping triggered). The training-validation gap of 4.81% (89.25% vs 84.44%) indicates our regularization strategy (L2, batch normalization, dropout) successfully prevented overfitting.

V. DISCUSSION

A. Why We Surpassed the Original Paper

Five methodological improvements contributed to our better performance:

1) **Proper shuffling:** Eliminated temporal leakage from sorted data
2) **Raw sensors only:** Avoided circular reasoning from engineered features
3) **Class weighting:** Forced the model to learn all three fatigue levels
4) **Stratified splitting:** Maintained class distribution in train/test sets
5) **Comprehensive regularization:** Prevented overfitting through multiple techniques

B. Limitations and Failure Modes

Level 3 recall (78%): Missing 22% of severe fatigue cases is problematic for safety-critical applications. Ideally, we'd want at least 95% recall for this class, even at the cost of more false alarms.

Level 2 precision (47%): Excessive false alarms could lead to alert fatigue, where medical staff begin ignoring warnings.

Sample size: We used only 3% of the full dataset due to computational constraints. Training on all 16 million samples might improve performance.

Temporal independence: Our model treats each sample independently. It doesn't capture how fatigue accumulates over time.

Limited generalization testing: We only validated on 2023 Hajj data. Performance on other years or mass gathering events remains unknown.

C. Deployment Considerations

For real-world deployment, we recommend:

Adjust decision thresholds: Lower the Level 3 threshold to catch more severe cases. Better to check someone unnecessarily than miss someone who's about to collapse.

Human-in-the-loop: Use the model as a screening tool that flags individuals for human assessment, not as an autonomous decision system.

Trend monitoring: Track fatigue changes over time. Someone whose readings have been climbing for an hour is more concerning than a single high value.

Personalization: Establish individual baselines. A 25-year-old athlete and a 70-year-old with cardiovascular disease require different thresholds.

D. Ethical Considerations

Privacy: Physiological data reveals sensitive medical information. Deployment requires strong encryption, secure storage, and informed consent.

Bias: Our model trained on 64 participants from 2023. It may not generalize well to underrepresented demographic groups.

Transparency: Medical personnel must understand the model's limitations. Blind trust in predictions could lead to dangerous outcomes.

VI. CONCLUSION

A. Summary

We successfully replicated and improved upon Al-Shaery et al.'s work. By identifying and fixing critical data issues (sorted data, engineered features) and applying rigorous methodology (shuffling, stratification, class weighting), we achieved 88.15% accuracy—2.88 points better than the original 85.27%.

Our key insight: simpler is often better. Seven raw sensors outperformed larger feature sets when data leakage was properly controlled.

B. Main Findings

1) Data sorting creates severe leakage (99% fake accuracy vs 88% real accuracy)
2) Raw sensors outperform engineered features when leakage is controlled
3) Class imbalance requires explicit handling through weighting
4) Simple models with proper regularization can achieve strong performance
5) Safety-critical applications need high recall for dangerous classes

C. Future Directions

Temporal modeling: Incorporate LSTMs or Transformers to capture how fatigue builds over time.

Full dataset training: Use cloud computing to train on all 16 million samples.

Multi-modal fusion: Integrate environmental data (temperature, humidity, crowd density).

Personalization: Develop adaptive models that learn individual baselines.

Real-time optimization: Optimize inference for deployment on wearable devices.

Cross-event validation: Test on marathons, festivals, and other mass gatherings.

Explainability: Implement attention mechanisms or SHAP values to show why predictions were made.

D. Practical Recommendations

For researchers and practitioners working with similar data:

1) Always shuffle time-series data before train/test splitting
2) Prefer raw sensor measurements over engineered features
3) Use stratified splitting for imbalanced datasets
4) Apply class weights to handle imbalance
5) Validate on truly held-out data
6) Examine per-class metrics, not just overall accuracy
7) Prioritize recall for safety-critical classes
8) Deploy with human oversight and clear escalation protocols

ACKNOWLEDGMENT

Thanks to the Kaggle community for making the Hajj Crowd Activity Prediction Dataset publicly available and to Google for providing free computational resources through Colab.

REFERENCES

[1] A. Al-Shaery, M. A. Rahman, and S. M. Alqahtani, "Predicting physical fatigue in Hajj pilgrims using wearable sensors and deep learning," *IEEE Access*, vol. 12, pp. 45231–45245, 2024.
[2] C. Catal, S. Tufekci, E. Pirmit, and G. Kocabag, "On the use of ensemble of classifiers for accelerometer-based activity recognition," *Applied Soft Computing*, vol. 37, pp. 1018–1022, 2015.
[3] A. Ashraf, "AHQAD: Arabic Healthcare Q&A Dataset," Kaggle, 2023.
[4] Z. Ziya, "Hajj and Umrah Crowd Management Dataset," Kaggle, 2023.
[5] R. Bhuiyan et al., "Hajj Crowd Dataset 2021," GitHub, 2021.
[6] M. Khosravi et al., "Health challenges of Iranian Hajj pilgrims: A decade of surveillance data (2013-2022)," *Frontiers in Public Health*, vol. 13, 2025.
[7] A. Hassan et al., "Respiratory infections among Egyptian Hajj and Umrah pilgrims in 2022," *Archives of Public Health*, vol. 81, no. 229, 2023.
[8] M. Al-Tawfiq et al., "Epidemiology of mortality during Hajj pilgrimage (2012-2017)," *Travel Medicine and Infectious Disease*, vol. 53, 2023.
[9] S. Alalal Dinah Ahmed, "Hajj Crowd Activity Prediction Dataset," Kaggle, 2023.