Hey Siri, am I Drunk?
Intoxication Detection Using Smartphone Accelerometers

Sumedh Panatula
sumedh@uw.edu

Kathryn Xiong
kathrx@uw.edu

1. Abstract

Smartphones provide a convenient, non-invasive means to detect alcohol intoxication via their embedded accelerometers. Traditional approaches rely on handcrafted feature engineering, which introduces preprocessing overhead and may overlook subtle motion cues. We propose an end-to-end deep learning pipeline that ingests raw accelerometer data and learns robust representations without manual feature design. Using the UCI Bar Crawl dataset [1], we compare Multi-Layer Perceptrons (MLPs) and Convolutional Neural Networks (CNNs) under the same train/validation splits as previous studies. Our initial models have shown results that far surpass previous studies, with a twelve-layer CNN that reaches over 98% accuracy. We further demonstrate that a compact four-layer CNN (with only 15K parameters) still achieves an accuracy greater than 80%, highlighting its suitability for real-time deployment on resource-constrained devices. These findings show that end-to-end models on raw sensor data can outperform feature-based methods and pave the way for accurate, low-latency intoxication monitoring on everyday smartphones.

2. Introduction

Alcohol intoxication detection is vital for applications ranging from personal safety apps to clinical monitoring. Conventional methods, such as breathalyzers and blood tests, while accurate, require specialized hardware and are not amenable to continuous passive monitoring. Hence, non-invasive sensing modalities—thermal imaging [3], speech analysis [5], and video-based gait assessment [4]—have gained traction. Yet these approaches often incur privacy concerns, high computational costs, or reliance on external sensors.

In contrast, smartphones provide built-in inertial sensors—accelerometers and gyroscopes—that can capture fine-grained motion patterns associated with intoxication-induced gait and postural instability. The UCI Bar Crawl dataset [1] pioneered this direction, achieving 73% accuracy using feature-engineered MLPs and CNNs. However, manual feature extraction adds latency, demands domain expertise, and may miss latent temporal dynamics.

This work explores end-to-end deep learning architectures that process raw 3-axis accelerometer time series, eliminating handcrafted stages. We systematically evaluate MLPs and CNNs under identical data splits, perform ablation studies on segment length, and measure on-device inference performance to ensure practical deployability. Our key contributions include:

- A detailed comparison of end-to-end MLP and CNN models for raw accelerometer-based intoxication detection.
- Ablation studies on input window duration and model complexity, quantifying trade-offs between accuracy, latency, and parameter count.

2.1. State of related work

As of the present, several groups have investigated ways to detect intoxication non-invasively. Koukiou et al. [3] demonstrated that thermal infrared imaging can be used to reliably capture alcohol-induced facial ”flushing” by measuring subtle temperature changes across the face. Similarly, Schuller et al, [5] evaluated speech differences between sober and intoxicated individuals, noting increased slurring and shifts in rhythm, which can be used as predictors of the speaker's state. More recently, Park et al. [4] applied a deep learning framework to analyze videos of participants walking, achieving high accuracy in distinguishing intoxicated from sober movements.

2.2. Problem statement

Existing approaches used in the Bar Crawl paper rely on preprocessed feature engineering. Doing so, although effective for post-hoc analysis, increases computational load and limits real-time inference. When placed on mobile devices, the model can drain battery life and limit the range of devices the model can be used on (for instance, smartwatches). Furthermore, feature engineering often relies on human intuition and assumes that the produced features can capture all meaningful patterns in the signals. As such, the accuracy of these features may suffer from changes in the dataset, i.e., new users or changes to the environment.

The existing approaches used in the prior bar Crawl study rely heavily on handcrafted feature extraction and preprocessing of the data. While this approach is effective for post-hoc analysis, this approach increases computational overhead and impedes real-time inferencing. When placed on mobile devices, this computation can drain battery life and limit the range of devices the model can be used on (for instance, smartwatches). Furthermore, feature engineering often relies on human intuition and assumes that the produced features can capture all meaningful patterns in the signals. As such, the accuracy of these features may potentially reduce model generalization when applied to new users or environments.

2.3. Unique insight

We propose a learning framework that operates directly on raw accelerometer time series data, eliminating the need for handcrafted feature extraction. By utilizing an end-to-end learning pipeline, a model could potentially capture minute patterns that the engineered features could overlook. Our hypothesis is that a model trained on minimally processed sensor data can achieve comparable or exceed prior attempts.

2.4. Technical challenges

By using raw accelerometer data as opposed to features, we introduce several challenges. One such challenge is the presence of noise due to collecting data from a handheld device. This becomes even more prevalent when considering each individual's unique drunken behaviors or differences in how each participant carries their phone. Another challenge is that the UCI Bar Crawl dataset comprises only 13 participants. This small sample size increases the risk of overfitting and may limit our ability to generalize for real-world use.

2.5. Experiment plan

Our plan is to evaluate convolutional neural networks (CNNs) and long short-term memory networks (LSTMs) to model the accelerometer data, aiming to identify movement patterns that distinguish sober from intoxicated behavior. We are particularly interested in testing whether shorter time windows (e.g. 2s, 4s, 8s) can still deliver accurate predictions, as this would improve the practicality of the system for real-time applications. Additionally, we are interested in exploring fine-tuning models for individual users to see if that can improve performance, dependent on time and the complexity of this challenge.

2.6. Expected outcome

We expect that working directly with raw accelerometer data, without using feature engineering, can still match or even outperform the accuracy found in the Bar Crawl paper. Ideally, we will show that smaller time windows are still effective, or alternatively, define a minimum period with the greatest accuracy. We are also curious to see if incorporating some level of personalization can be more effective. We hope to see that a model tuned using a smaller set of samples for a certain individual can be more accurate in detection compared to a generalized model. Overall, our goal is to explore the limits of these approaches when operating in these less-than-perfect conditions.

3. Methods

3.1. Data preprocessing

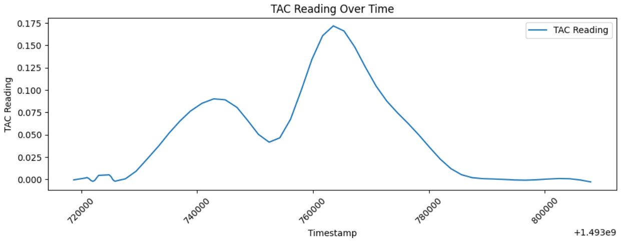

Figure 1. TAC readings for a single participant

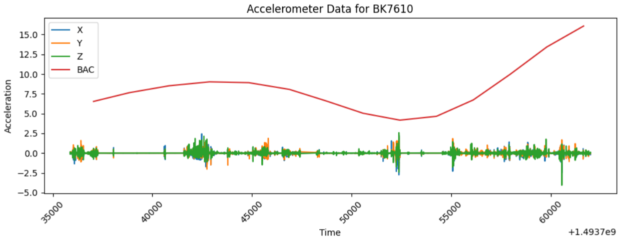

Figure 2. Corresponding accelerometer signals for a participant

The UCI Bar Crawl dataset was used, which includes both accelerometer data and TAC blood alcohol readings. This dataset already includes preliminary preprocessing steps, notably that it filters the accelerometer signals to reduce noise and minor fluctuations in the readings. Additionally, it features data in a standardized format, providing consistent units and timestamps to allow for easier parsing and integration into models.

From there, we applied further preprocessing to address gaps and duplicate timestamps in the accelerometer data as well as differences in polling rates between the TAC and accelerometer readings. To address these issues, we utilized a strategy modeled after the one described in the original Bar Crawl study. Specifically, for each participant's accelerometer data, we isolated periods of non-zero movement–defined as intervals where the vector magnitude of acceleration exceeded a baseline threshold. Consecutive segments that satisfied this criterion were appended together to form an aggregate, and this aggregate was retained if it exceeded ten seconds. Then, in order to match the 40 Hz sampling rate of the accelerometer data to the TAC readings collected every 30 minutes, we linearly interpolated each segment to follow the original 40 Hz sampling and aligned the segment to the nearest real-time TAC reading. By doing so, we were able to approximate corresponding TAC readings for each segment.

3.2. Model Architectures

Our goal was to meet or exceed the Bar Crawl paper's reported accuracies (73.28% for their MLP, 72.47% for their CNN) by designing deeper, more expressive models. We implemented and compared the following architectures:

Multi-Layer Perceptron (MLP)

Starting from the baseline shallow MLP (single hidden layer) used in the original study, we explored progressively deeper variants. Early experiments started with three hidden layers (each 64-128 units, with Leaky ReLU activations), which yielded only 60% accuracy. To identify better configurations, we leveraged Optuna [2] for automated hyperparameter optimization. Completing over 300+ trials, the resulting best model consists of five hidden layers (units per layer: 156-198-59-51-33) and achieved 80% accuracy, surpassing even the best result from the Bar Crawl paper. Wondering whether we were correctly capturing the input with such small first-layer sizes, we expanded to a deeper architecture of 1024–1024–512–256–128, which ultimately achieved 93.3% accuracy.

Convolutional Neural Network (CNN)

Building upon the paper's two-layer CNN, we progressively increased depth and monitored validation accuracy with inputs of size $N \times 3 \times 400$. Each convolutional block consisted of a 3×3 convolution followed by a Leaky ReLU, with max-pooling applied after every two blocks. These feature maps were then flattened, passed through a dropout later ($p = 0.5$), and fed into a linear layer.

Early trials with CNNs incorporated batch normalization layers; however, this consistently yielded lower validation accuracies. We hypothesize that the added normalization may have resulted in over-regularization, hindering the model's capacity for learning meaningful patterns. As a result, we shifted our focus to designing deeper models.

As we explored more complex models, we saw initial success with an architecture comprising of four convolutional blocks (kernel size of 3, 64 filters for all stages). This model attained an 84% validation accuracy, and as a next step, we reduced the number of filters in the intermediary layers (reducing filter size to 16 for the first 3 layers). This second iteration of the model saw very similar accuracy, with only a 0.2% difference in overall accuracy. This indicated to us that the model was unable to capture enough given its small size, as such we expanded to a six-layer network(filter sizes being 8-16-32-64-128-256), which saw an outstanding 95.61% validation accuracy. As a further push, we explored a doubling of our model complexity, pushing towards a twelve-layer convolutional network (filter size of 64 throughout), which resulted in an accuracy of 98%. These results show a clear trend: as we deepen the network, the model better learns nuanced patterns and is more accurate at predicting intoxication.

3.3. Results

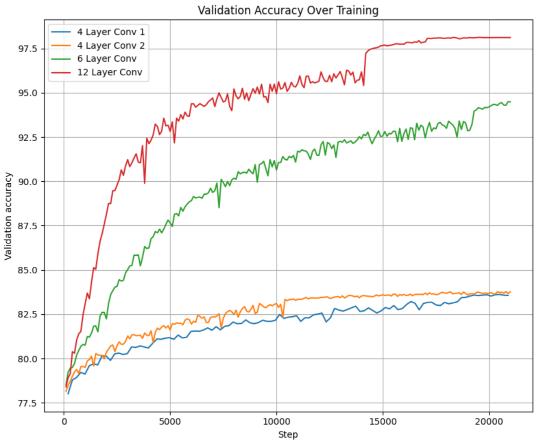

Figure 3. Model validation accuracies

Model	Parameters	Val. Accuracy
12-layer ConvNet	136897	98.12%
6-layer ConvNet	144321	95.61%
4-layer ConvNet (iter. 1)	44097	84.19%
4-layer ConvNet (iter. 2)	15121	83.93%

Table 1. Model parameters and accuracies

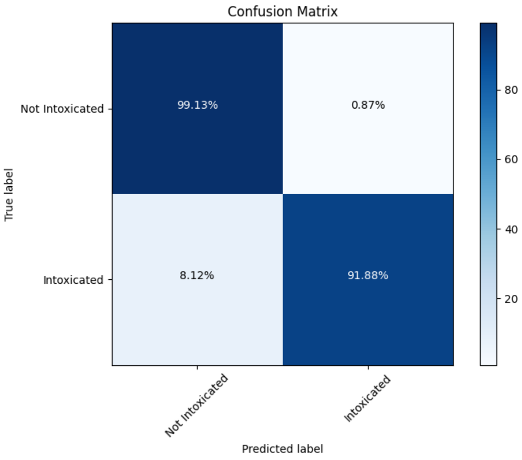

Figure 4. Accuracy of the 12 Layer ConvNet

As shown in the figure above, our twelve-layer CNN maintains an exceptionally low false positive rate, under 1%, when classifying non-intoxicated data inputs. However, it exhibits a higher false negative rate of approximately 8%, meaning that in roughly 8% of intoxicated input data, the model fails to flag drinking behavior.

4. Discussion

4.1. Model Portability

A notable outcome of this study is the potential for real-time intoxication monitoring using handheld devices. Given the widespread presence of smartphones and their built-in accelerometers, the ability to run models on phones allows for accessible and reliable intoxication inferences.

To evaluate the portability of each model, we asses the accuracies and the number of parameters for each model. Our smallest model, the 4-layer ConvNet (iteration 2), contains only 15121 parameters and still achieves roughly 80% validation accuracy. This makes it very lightweight and well-suited for mobile deployment with minimal effect on battery life.

In comparison, our longest model, the 12-layer ConvNet, scores the highest accuracy of 98%, but consists of nearly 140,000 parameters. Though considerably larger, this model is still within the capabilities of modern smartphones. However, its larger architecture leads to greater power consumption and inference time, which makes it less practical for continuous use.

4.2. Varied Window Sizes

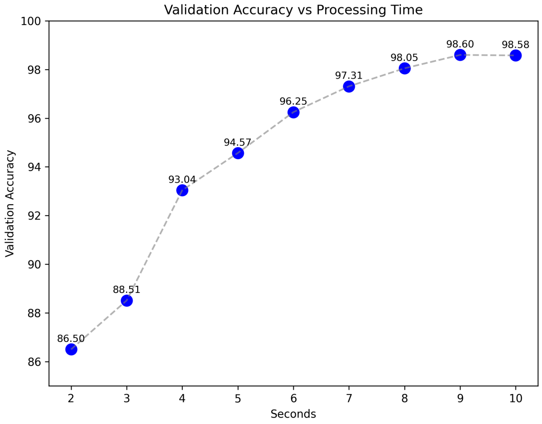

Figure 5. ConvNet accuracy with varying input segment sizes

We further evaluated the impact of segment duration on model performance by varying the window sizes of segments inputted into the model. The goal was to identify the shortest window that could maintain high predictive accuracy, which would enable more efficient inference. For this, we used our current best-performing model, the 12-layer ConvNet.

As shown in Figure 5, very short segments (2-3 seconds) yielded noticeably lower accuracies. From 4-7 seconds, the accuracies increased considerably and continue to increase. After 7 seconds, the accuracies appear to plateau at around 98%.

These findings are particularly meaningful for real-time deployment, where shorter windows can reduce resource consumption and prediction time. Since windows after 7 seconds provide negligible accuracy gains while increasing memory usage, using a 7 second window would represent an effective balance between model performance an computational cost.

4.3. Segment Overlap

One experiment we conducted was evaluating performance differences when using overlapping segments compared to non-overlapping ones. For overlapping segments, we used a sliding window to generate multiple n-second segments that may share timestamps with other segments. Using overlapping segments allowed for more continuous data coverage, which allowed the model to fully capture and respond to intermediate behavior patterns in a participant's movement.

In contrast, training with non-overlapping segments limited the model's ability to learn these intermediate behaviors, so the model struggled to predict when presented with these intermediate behaviors at test-time, resulting in accuracies in the low 80% range.

4.4. Live Testing

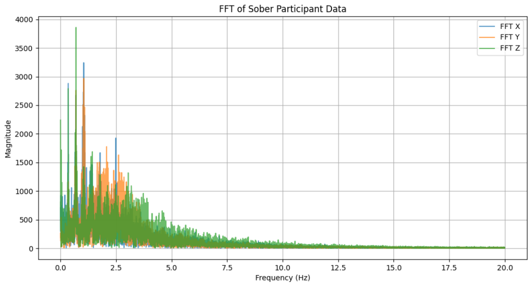

Figure 6. Fourier Transform of Sober participant data

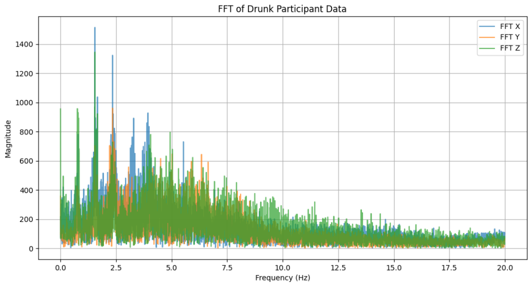

Figure 7. Fourier Transform of Intoxicated participant data

We also decided to evaluate the performance of the model in real-time. To do so, we had participants walk while holding a smartphone which was taking accelerometer readings at 40 Hz. We first had them walk for two minutes prior to in-taking any alcohol. Afterwards, we repeated this procedure twice for every 360 mL in-taken.

To better understand the underlying patterns in the collected data, we analyzed the frequency characteristics of accelerometer signals using the Fast Fourier Transform (FFT). Figure 6 shows an FFT for the walking data of a participant when they are sober. Notably, the greatest frequency occurrences are concentrated around 2.0 Hz, corresponding to a natural walking rhythm and pattern. Figure 7 shows an FFT for the walking data of that same participant after consuming 3 drinks. The frequencies in this graph are much more scattered and irregular, reflecting the erratic gait typically associated with intoxication.

In our evaluation, we observed a drop in accuracy compared to results reported on the original dataset. This performance gap may stem from two primary factors: the limited size of our training data and potential discrepancies in the data collection process. Since the original dataset included only 13 participants, it's possible our model overfitted to individual-specific features, limiting generalizability. Additionally, without access to the exact data collection methods used in the original study, there's a risk that differences in format or quality impacted model performance.

One notable observation came from analyzing the model's raw logit outputs: for sober data, the average logit was -34, while intoxicated samples averaged -6. This suggests the model does distinguish between the two classes, but with low confidence—even in intoxicated cases, the outputs remain weakly negative.

5. Future Studies

Future work could focus on collecting more comprehensive and meaningful data to improve model accuracy and generalization. This would involve significantly expanding the participant pool beyond that of the original study of 13 participants monitored over a 24-hour period. The added data should focus on acquiring data both from more participants and over multiple drinking occasions, allowing for a wider range of drinking behaviors and responses.

Additionally, new data could incorporate more descriptive details about each drinking event. This would include when the participant drank as well as details about what they drank, such as the alcohol content of the drink and the intake at a time. Doing so would allow for better interpolation of TAC levels in between readings.

References

[1] Bar Crawl: Detecting Heavy Drinking. UCI Machine Learning Repository, 2020. DOI: https://doi.org/10.24432/C5TK6G. 1

[2] Takuya Akiba, Shotaro Sano, Toshihiko Yanase, Takeru Ohta, and Masanori Koyama. Optuna: A next-generation hyperparameter optimization framework. *CoRR*, abs/1907.10902, 2019. 3

[3] G. Koukiou, G. Panagopoulos, and V. Anastassopoulos. Drunk person identification using thermal infrared images. In *2009 16th International Conference on Digital Signal Processing*, pages 1–4, 2009. 1

[4] Suah Park, Byunghoon Bae, Kyungmin Kang, Hyunjee Kim, Mi Song Nam, Jumyung Um, and Yun Jung Heo. A deep-learning approach for identifying a drunk person using gait recognition. *Applied Sciences*, 13(3), 2023. 1

[5] Björn Schuller, Stefan Steidl, Anton Batliner, Florian Schiel, Jarek Krajewski, Felix Weninger, and Florian Eyben. Medium-term speaker states—a review on intoxication, sleepiness and the first challenge. *Computer Speech Language*, 28(2):346–374, 2014. 1