Multimodal Action Classifier

A look into the performance of various classifiers in multimodal action recognition

Ruben Alias
Electrical and Computer Engineering
Rutgers University
New Jersey
rva16@scarletmail.rutgers.edu

Romany Ebrhem
Electrical and Computer Engineering
Rutgers University
New Jersey
re277@scarletmail.rutgers.edu

ABSTRACT

This paper provides an extensive analysis of the use of the plethora of smartphone sensors for the real-time Classification of human actions, intending to enhance the monitoring of older people's health and safety. Using phone-integrated accelerometers, gyroscopes, and gravity, we trained a machine learning that can accurately classify dynamic human actions. Our research compared several machine learning models, including Convolutional Neural Networks (CNN), Long Short-Term Memory networks (LSTM), and Transformer models, focusing on optimizing accuracy. With an outstanding accuracy of 96.95% and higher resilience in live testing, we concluded that the Transformer performs the best out of the three models tested. We discuss the intersection of our research with applications within the healthcare industry, particularly in dealing with old-age patients concerning monitoring behavior in real-time.

1 Introduction

In the digital era, the spread of smartphones with various sensors has unlocked new frontiers in data collection and analysis. These ubiquitous devices' integrated acceleration, gyroscope, and gravity sensors offer a fertile ground for innovative health monitoring solutions. The elderly population stands to benefit significantly from advancing real-time activity monitoring. This paper explores the benefits of using smartphone sensors to classify human movements, which can be used for many applications, such as health monitoring for the elderly.

The motivation for this research arises from a need to develop cost-effective methods for activity tracking. Current solutions often require wearable sensors, which may not be accessible to the general population or cause discomfort when worn. As in modern-day society, almost everyone is walking around with a smartphone, which carries many sensors that can be efficiently utilized. By utilizing smartphone sensors, we aim to generate a machine-learning model that can differentiate between activities, with the ultimate goal of providing real-time feedback and enhancing the quality of health monitoring for senior individuals.

This research paper aims to advance the class of activity classification by approaching different problems, such as aligning and understanding different multimodal sensor data. Throughout this paper, we will discuss related work that has been done in this field as well as different approaches to this problem that have been previously attempted. We will also discuss the data collection and the data alignment process, emphasizing our methodology and how it outperforms previous work. We will go into great detail about the outcome and how it validates the robustness and dependability of our system. As we wrap out the study, we continue discussing potential directions for further research.

2 Related Work

Our research was significantly influenced by the literature "Multimodal Classification: Current Landscape, Taxonomy, and Future Directions.". This paper introduces a new taxonomy for multimodal classification, which helps set standard terminology in the machine learning field. This research also gives a pervasive analysis of multimodal classification models compared to unimodal models, finding great promise in the advancement of multimodal in outperforming unimodal. They put much emphasis on how there are still many issues with multimodal that need to be figured out, such as big data, class imbalance, and instance-level difficulty. Instead of comparing multimodal to unimodal, as the paper has done, we compared multimodal to multimodal modal but still need a modality. There have been many advancements since this paper's release, and we are confident that multimodal performs better than unimodal, so our question relies on whether it is still reliable when missing a modality.

The previous work in "Smartphone motion sensor-based complex human activity identification

using deep stacked autoencoder algorithm for enhanced smart healthcare system" greatly impacted our technology use. Similar to this research paper, our primary goal was establishing an action classifier with smartphone sensors. In our research, we use multimodal sensors such as acceleration, gravity, and gyroscope data; however, in this paper, they only use acceleration to achieve this task. Both papers focused on monitoring health status, fall detection, and estimating energy expenditure. In their paper, they used an Autoencoder algorithm to achieve their results with multiple acceleration components to help resolve the effects of orientation inconsistencies. We used this method when approaching our multimodal modal while adding on other sensors to increase accuracy.

From the paper "Real-time Smartphone Activity Classification Using Inertial Sensors—Recognition of Scrolling, Typing, and Watching Videos While Sitting or Walking," we conclude by using Real-time Smartphone sensor data in our research. Their focus was using the on-phone IMU sensor for activity classification, such as scrolling, typing, and watching videos. They achieved an accuracy of 78.6% with the Extremely Randomized Trees algorithm. While these results are significant, we achieved 96% using a machine learning transform model to classify waking, idling, and jumping. While their objective is to enable applications to receive data from the phone's sensors for improved personalization, our case use was for health benefits. However, their methodology of gathering the sensor data in real time and sending and receiving it from a server set the backbone of our multimodal action classifier. We utilized a TCP socket to complete the connection between the smartphone and the host computer, running the machine learning algorithm.

3 Data/System Description

Smartphones, containing many integrated sensors, have evolved into personal data centers that continuously collect data. The three primary sensors used in this research are the accelerometer, gyroscope, and gravity sensor. With TCP sockets connecting these sensors to our central system, real-time data gathering and processing are accomplished with the shortest latency possible. This data offers an extensive representation of the user's motions and orientations.

3.1 Sensors Utilized

Three essential sensors found in modern smartphones form the basis of our data collection system:

Accelerometer: This sensor measures acceleration, which is the change in velocity over time $a=\Delta v/\Delta t$. The data collected from the acceleration is vital as it is the base of movement on which the machine learning model can train. Since it is Vital to detect linear acceleration, we collected the X and Y components of the acceleration. This allows us to distinguish between running, walking, and idling. These points provide a person's inertial erection and its motion in a three-dimensional space. The accelerometer provides advantages such as reliability and the ability to work with other sensors.

Gyroscope: In addition to the accelerometer, the gyroscope measures the rotational rate along the three axes of the smartphone. To distinguish between subtle motions and directional changes, this sensor is crucial for registering orientation and rotational movements. This sensor was a great addition to the accelerometer as it built on it. In future testing, the gyroscope will be crucial for fall detection for the safety of individuals. For our particular use case, we only consider the Z component from the gyroscope's data, as the Z-Coordinate corresponds to a rotation vector through the flat side of the phone(where the most significant rotation occurs on the device per a given action while it is in the pocket.

Gravitometer: The gravity sensor helps determine the device's orientation concerning the Earth's surface by separating the force of gravity from other accelerative motions. This data is beneficial in determining the user's stance but, more importantly, the position of the subject's legs at a given moment.

3.2 Data Collection Process

Sensor Logger, a mobile application, is the source of the data streaming process. This logger is the main hub for gathering and sending sensor data in real-time. A Flask server accepts and processes real-time sensor information, facilitating the integration. This server creates a smooth connection by coordinating data exchange between the machine-learning model and the smartphone sensors. Through communication with this logger, the Flask server obtains a steady stream of data that guarantees the machine learning model is updated with the most recent movement data about the user.

A master data frame combines several datasets according to timestamp values to enable multimodal processing. By ensuring synchronization across modalities, this stage enables a thorough comprehension of the user's actions. Two methods handle missing values: zero insertion and forward filling. We first forward fill any existing values. However, initial NaNs created by NumPy's library functions are then substituted with 0s.

Then, labeled samples are created for testing and training using the master data frame. A TensorDataset is created for PyTorch compatibility once samples are collected for every action label. It is to be noted we did not normalize values as it did not seem to impact performance.

Three machine learning models—LSTM, CNN, and Transformer—were applied after the data-collecting stage. The main goal was to find the best configurations that produced the maximum accuracy by carefully adjusting the hyperparameters. After that, each model's performance was thoroughly assessed, focusing on measures related to accuracy and stability. The model that demonstrated the highest level of accuracy in addition to solid stability was the transformer. It is worth noting that these particular models will be further examined in later portions of this report.

4 Objectives/Hypothesis

The objective is to develop a machine-learning model that can classify human activities based on sensor data, specifically focusing on applications for the elderly. We hypothesize that a machine learning model can be trained to classify daily activities with high accuracy, thereby enhancing the quality of life and safety of the elderly.

The primary motivation for the research paper is to assess the impact of removing mobility data from a machine learning Convolutional Neural Network (CNN) on the accuracy of the classifier model. It is crucial to understand the effects of removing modality from a model because, in deployed applications, a sensor may malfunction. Therefore, it must ensure that the model is performing with high accuracy. Also, we aim to identify which machine learning model can perform the best in terms of accuracy: LSTM, CNN, or Transformer. By fine-tuning each model's hyperparameter, we can achieve the highest accuracy with an overarching objective of developing a system utilizing smartphone sensors to monitor the movements of the elderly in their daily lives to enhance health and safety.

5 Feature Selection and Engineering

Effective feature selection is critical to the model's success. This process begins with aligning sensor data, considering the varying frequency and amplitude scales. Sensor alignment occurred by combining the different modalities' time columns in the Pandas Dataframe in which each was stored. Following this, the timestamps were sorted and used as a basis for the new data frame. Subsequently, all data was inserted into the new data frame based on each modality's timestamps.

Following this, it was essential to address the issue of specific modalities coming in at different timings and frequencies. Hence, we upsampled the data by propagating values until the next ‘updated’ value occurred. Given that no modality was tested before our final setup was updated at a rate slower than 1Hz, we found this a suitable method.

6 Multimodal Data Patterns

This section explores the unique patterns that emerge from the multimodal nature of the data collected. We address the challenges in data alignment and the strategic methodologies employed to interpret complex sensor signals. We aim to identify distinct activity signatures that our models can accurately classify.

6.1 Idling

In this section, we observe that, as expected of data collected on an idle user (barring any anomalies), the sensors read an approximately idle state on all modalities. Given how distinct the idle state is concerning the other modalities, classifying this particular action had the highest success rate in live testing.

6.2 Walking

Across the accelerometer and gyroscope, we observe that the sensor readings oscillate(as expected when collecting data on walking) within a fixed range and almost exhibit sinusoidal-like behavior. We also note that the gravitometer readings for X and Z showed an evident oscillation pattern.

6.3 Jumping

Although the data collected in Jumping is also oscillatory, we notice notable consistent spikes across the Y axis on the accelerometer, which we believe the model will learn to pick up on. Furthermore, the Z-axis rotation on the gyroscope is also far more prominent in this action class. Finally, we observed a drift in the gravity vector during data collection. We attribute this to the sensor moving during training.

7 Architectures, Experiments, and Methodology

7.1 Architectures

For selecting the most efficacious model for the given task, we considered 3 Deep Learning models: a Transformer Encoder, CNN, and LSTM. We go into the specifics of the reasonings behind each choice and the parameters selected.

7.1.1 Transformer Encoder Array

Following the success of the Transformer Model in sequence modeling from the “Attention is All You Need” paper, we attempted to use the transformer architecture

in our experimental setup since we are also dealing with data where sequence matters. However, since we are only dealing with creating a class-based representation of the input signals, the decoder portion of the full transformer model was deemed unnecessary.

The specific choice of 5 Transformer Encoder Layers was found via testing, as they required fewer Epochs to train and did not lead to overfitting on our training data. We also hit upon a learning rate of .0001 in the hopes of a small learning rate, allowing the model to more accurately hit the global minimum of the cost curve and to prevent oscillations in the loss.

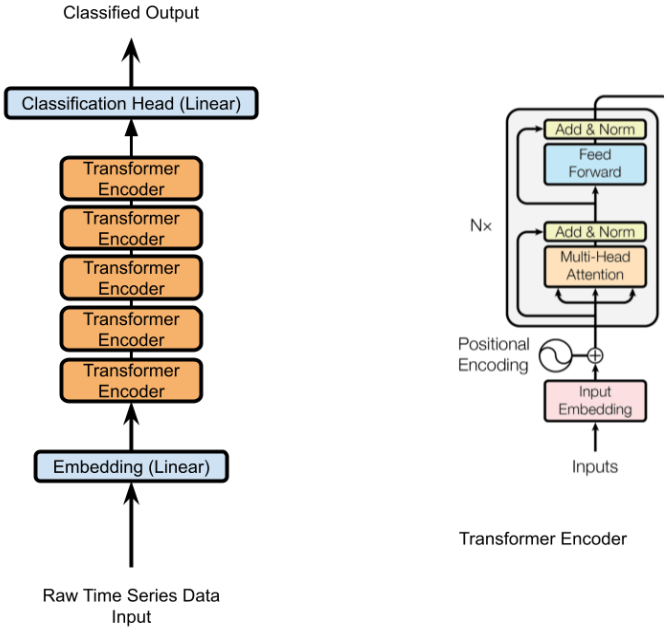

Figure 1: Transformer Encoder Array with classification head (left) Transformer Encoder architecture (right)

7.1.2 Convolutional Neural Network

Convolutional Neural Networks are frequently used in Image Classification because they extract local patterns as intermediate features and representations. Given that our data is inputted as time series, and we assume that different classes of actions will have distinct local patterns in time across the various modalities that the convolutional filters could extract, it seemed like a suitable choice for the task of Time Series Classification.

The parameters were chosen based on results from experimentation. In particular, because of the relatively small size of our dataset and how controlled our setup is, we chose not to make the network as deep as many other CNN-based classification models.

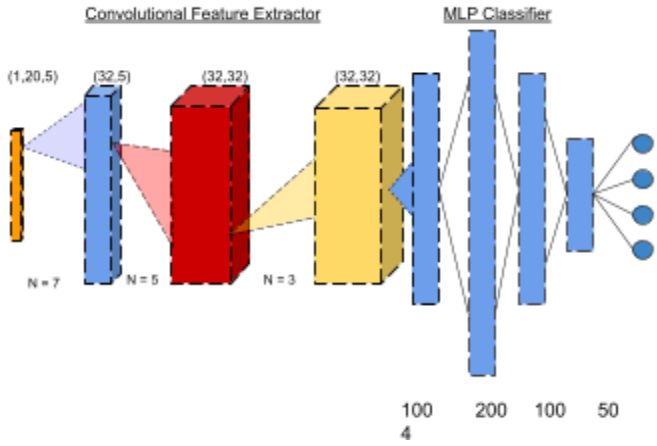

Figure 2: CNN Architecture with Convolutional Layers(left) and MLP Classifier(right) with parameters for each listed. Above each convolutional layer are the layer dimensions, below are the filter sizes, and below the classifier layers are the number of units per fully connected layer.

7.1.3 Long Short-Term Memory

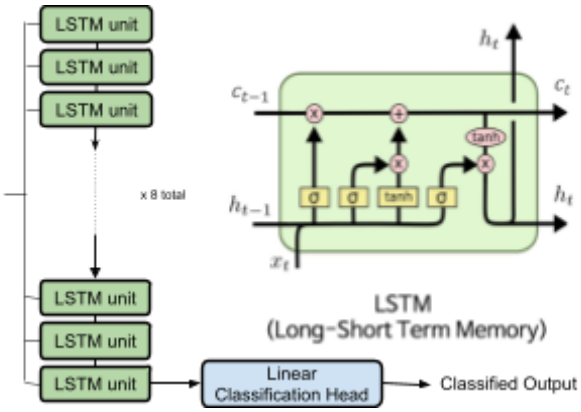

Figure 3: LSTM Array (left) connected to a Linear Classification head (bottom) and an example of the architecture of a single LSTM unit (right).

The LSTM model is a staple of dealing with time series data and is the ancestor of the transformer model, historically used for sequence-to-sequence modeling. Just as with the CNN, the number of layers in our LSTM array was chosen

based on testing how well it trains from the given data in our dataset.

7.2 Experiments and Methodology

Our experiment compared the various architectures and their performance on multimodal time-series data. Our methodology for comparison was to observe the model performance on the testing data collected from a separate location from the training data, test the live behavior of each model, and subjectively evaluate the model's performance.

Furthermore, we created an additional test to assess each model's few-shot performance to test the capability of Multimodal Learning in few-shot testing. Using a similar test loop to our base testing setup, we add functionality where 1 of 5 of the inputs is set to 0 randomly during each test. Given that this is randomly distributed, we assume the results from this test are a sufficient representation of the average performance of each model on Few Shot testing across all modalities.

8 **Results**

8.1 Multimodal best case testing

The results section presents a detailed analysis of each model's performance. We report on accuracy and testing loss, providing insights into the strengths and weaknesses of each approach. The Transformer model demonstrates exceptional performance, suggesting its suitability for sequence-based sensor data interpretation.

Method	Accuracy	Loss
CNN	91.77%	0.1024
LSTM	91.16%	0.4898
Transformer	96.95%	0.3861

The CNN and LSTM perform approximately the same on the testing data.

Upon testing on live data, we found that the Transformer also appears to have a far more consistently accurate output than the CNN and LSTM. However, across all three classes, there seems to be difficulty in making correct inferences regarding jumps. This may be attributed to errors in the experimental setup or data collection. More experimentation would be required to improve the accuracy of this class.

8.2 Multimodal Few Shot Testing

This section explores our Few Shot performance testing results mentioned in prior sections.

Method	Accuracy	Loss
CNN	69.21%	3.410
LSTM	66.46%	1.8933
Transformer	78.66%	.8613

We find from testing each model that the Transformer is again the most successful in the test, followed by the CNN and the LSTM. We especially note that the difference in performance between the CNN and LSTM is more apparent in this test, perhaps hinting at CNN's ability to pick out local patterns via filtering, which leads to better performance across modalities.

9 **Discussion**

We interpret the results in the discussion in light of our original objectives and hypotheses. The practical implications of the findings are considered, including the potential to replace wearable sensors with a more seamless smartphone-based system. We also explore the challenges faced during the study and the limitations of our approach.

We can conclude that the transformer machine learning model proved to be the most accurate and stable, especially in Few Shot performance, and that further testing must be done on this model. In training the Transformer model, we used the data set we produced from our real-life experiments. This was sufficient for our purpose, but to scale this research further, we will need a plethora of data to train the model before using it on real test subjects to gain meaningful results. Once we can obtain an ample data set, we can start testing this technology on real subjects or anyone who wishes to track their activity data. For our use case, we market this toward elderly people who want to track their movement daily without using smartwatches. This allows for real-time monitoring, which could help with

fall detection and ensure the user gets ample daily movements. This could be used to show healthcare providers a patient's activity to help assess their help.

10 Conclusion

In conclusion, after testing Long Short-Term Memory (LSTM), Convolutional Neural Networks (CNNs), and a Transformer model, we found that the transformer outperformed all the other models with an astounding 96.95% accuracy with a testing loss of only 0.1024. The transformer model proved to be the most accurate and had the best stability when training the model. We also found that the CNN model learned faster while reaching 91.77% accuracy, which may be more efficient with a larger data set. The next step in this research is to obtain more data on different types of actions to expand the number of classes the model can detect, making it more usable in a real-life scenario. Once more data is retrieved, we can train the model on this new data and begin testing human testing.

REFERENCES

[1] PyTorch. "Training a Classifier¶." Training a Classifier - PyTorch Tutorials 2.2.0+cu121 Documentation, pytorch.org/tutorials/beginner/blitz/cifar10_tutorial.html. Accessed 10 Dec. 2023.

[2] Zongbo Han, Changqing Zhang, Huazhu Fu, and Joey Tianyi Zhou. Trusted multi-view classification. arXiv preprint arXiv:2102.02051, 2021

[3] https://docs.google.com/presentation/d/1GWG5Lz9Q_zwXdJDAdSGKYFjB QoKiMZbgCoMSZu0h--M/edit?usp=sharing

[4] Zhuo S, Sherlock L, Dobbie G, Koh YS, Russello G, Lottridge D. Real-time Smartphone Activity Classification Using Inertial Sensors—Recognition of Scrolling, Typing, and Watching Videos While Sitting or Walking. Sensors. 2020; 20(3):655. https://doi.org/10.3390/s20030655

[5] Alo UR, Nweke HF, Teh YW, Murtaza G. Smartphone Motion Sensor-Based Complex Human Activity Identification Using Deep Stacked Autoencoder Algorithm for Enhanced Smart Healthcare System. Sensors. 2020; 20(21):6300. https://doi.org/10.3390/s20216300

[6] C. Sleeman Virginia Commonwealth University, William, et al. "Multimodal Classification: Current Landscape, Taxonomy and Future Directions." *ACM Computing Surveys*, 1 July 2023, dl.acm.org/doi/10.1145/3543848.