Activity Recognition Using Machine Learning Techniques

ECE 5258

Alesandra Olinde Vergara

Florida Institute of Technology
https://github.com/sasha-ov/ECE5258-ICP

December 12, 2024

Outline

- Problem of Interest: Human Activity Recognition (HAR)
  - Importance and applications
- Dataset
  - Human Activity Recognition Using Smartphones Dataset
  - Types of activities and sensor data
- Time-Series Data and Neural Networks
- Baseline Model
  - Support Vector Machine (SVM)
  - Performance and limitations
- Proposed Models
  - CNN, LSTM, CNN-LSTM, and TCN architectures
  - Training, validation, and test performance
- Comparative Analysis
  - Confusion matrices and classification reports
  - Strengths and weaknesses of each model
- Conclusion and Future Work
  - Key findings
  - Recommendations for improvement

Problem of Interest: Human Activity Recognition Using Neural Networks

Human Activity Recognition (HAR) is the process of analyzing sensor data to determine the physical activity being performed by an individual.

This project investigates the application of *Neural Networks (CNNs, LSTMs, TCNs)* to the HAR problem, focusing on improving classification accuracy for challenging activities like sitting and standing.

Examples of HAR Applications:

- **Healthcare Monitoring**
  - Tracking patient mobility
- **Fitness and Wellness Apps**
  - Providing activity insights for users

For more information, see the Papers With Code HAR.

Data Resource

Dataset The dataset used is the *Human Activity Recognition Using Smartphones Dataset*, which includes inertial sensor data (accelerometer and gyroscope) recorded at 50 Hz from 30 participants performing six activities:

1. Walking
2. Walking Upstairs
3. Walking Downstairs
4. Sitting
5. Standing
6. Laying Down

The dataset is loaded and processed as raw signal data, with no feature extraction, using custom utility functions for direct analysis with Neural Networks.

Dataset Reference: Human Activity Recognition Using Smartphones Dataset on UCI Archive.

Time-Series Data Processing with Neural Networks

Neural networks offer versatile methods for time-series analysis:

- **Convolutional Neural Networks (CNNs):** Use 1D convolutions to extract local temporal patterns, with pooling layers for feature reduction. Advantages:
  - Computationally efficient and effective for short-term dependencies.
- **Long Short-Term Memory (LSTMs):** Leverage gated mechanisms to store long-term dependencies and address the vanishing gradient problem. Advantages:
  - Capture both short- and long-term temporal relationships.
- **Temporal Convolutional Networks (TCNs):** Employ causal and dilated convolutions for long-term dependencies, ensuring no future leakage. Advantages:
  - Parallelizable and scalable for both short- and long-range patterns.

Key References: Wang et al. (2017), Hochreiter Schmidhuber (1997), Bai et al. (2018).

Baseline Model: Support Vector Machine (SVM)

Baseline Model Overview and Performance:

- **Type:** Multiclass SVM with One-vs-All (OVA) strategy.
- **Kernel:** Gaussian (Radial Basis Function), tuned via 10-fold cross-validation.
- **Features:** 561 time and frequency domain characteristics from raw signals.

Performance:

- Overall accuracy: 96%.
- Lowest recall: Sitting (88%), struggles with static activities.

	WK	WU	WD	ST	SD	LD	**Recall**
Walking	**492**	1	3	0	0	0	99%
W. Upstairs	18	**451**	2	0	0	0	96%
W. Downstairs	4	6	**410**	0	0	0	98%
Sitting	0	2	0	**432**	57	0	88%
Standing	0	0	0	14	**518**	0	97%
Laying Down	0	0	0	0	0	**537**	100%
Precision	96%	98%	99%	97%	90%	100%	**96%**

Figure 1: Confusion Matrix of SVM.

Model Architectures

- **CNN:**
  - 3 Convolutional layers with MaxPooling and Dropout.
  - Dense layer with 128 neurons.
  - Softmax output layer.
- **LSTM:**
  - 2 LSTM layers with Batch Normalization and Dropout.
  - Dense layer with 64 neurons.
  - Softmax output layer.
- **TCN:**
  - TCN layer with multiple dilations and skip connections.
  - Dense layer with 64 neurons.
  - Softmax output layer.

Training and Validation History

- **CNN:** Final Training Accuracy: ~97.38%, Highest Validation Accuracy: ~92.74%.
- **LSTM:** Final Training Accuracy: ~95.81%, Highest Validation Accuracy: ~93.65%.
- **TCN:** Final Training Accuracy: ~95.28%, Highest Validation Accuracy: ~93.01%.

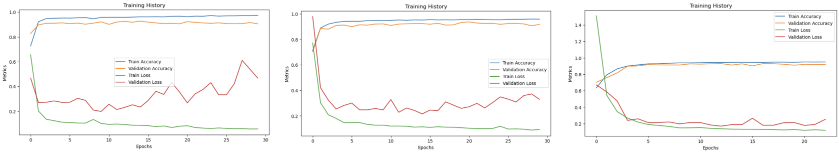

Figure 2: Training and Validation Histories for CNN (left), LSTM (middle), and TCN (right)..

Model Evaluation

- **CNN:**
  - Test Accuracy: 92.16%, Test Loss: 0.2547.
- **LSTM:**
  - Test Accuracy: 92.37%, Test Loss: 0.2758.
- **TCN:**
  - Test Accuracy: 93.01%, Test Loss: 0.2381.

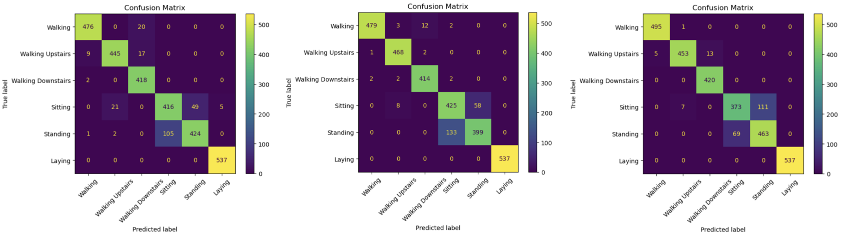

Figure 3: Confusion Matrices for CNN (left), LSTM (middle), and TCN (right).

CNN Classification Report

Class	Precision	Recall	F1-Score
Walking	0.98	0.96	0.97
Walking Upstairs	0.95	0.94	0.95
Walking Downstairs	0.92	1.00	0.96
Sitting	0.80	0.85	0.82
Standing	0.90	0.80	0.84
Laying	0.99	1.00	1.00
Accuracy		**0.92**	
Macro Avg	0.92	0.92	0.92
Weighted Avg	0.92	0.92	0.92

Table 1: CNN Classification Report

LSTM Classification Report

Class	Precision	Recall	F1-Score
Walking	0.99	0.97	0.98
Walking Upstairs	0.97	0.99	0.98
Walking Downstairs	0.97	0.99	0.98
Sitting	0.76	0.87	0.81
Standing	0.87	0.75	0.81
Laying	1.00	1.00	1.00
Accuracy		**0.92**	
Macro Avg	0.93	0.93	0.93
Weighted Avg	0.93	0.92	0.92

Table 2: LSTM Classification Report

TCN Classification Report

Class	Precision	Recall	F1-Score
Walking	0.99	1.00	0.99
Walking Upstairs	0.98	0.96	0.97
Walking Downstairs	0.97	1.00	0.98
Sitting	0.84	0.76	0.80
Standing	0.81	0.87	0.84
Laying	1.00	1.00	1.00
Accuracy		**0.93**	
Macro Avg	0.93	0.93	0.93
Weighted Avg	0.93	0.93	0.93

Table 3: TCN Classification Report

Classification Report: CNN, LSTM, and TCN

Model Comparisons for Precision, Recall, and F1-Score:

- **CNN:** Balanced performance, but struggles with sitting and standing.
- **LSTM:** Slightly better long-term dependency handling; consistent high performance.
- **TCN:** Best overall results, particularly in dynamic activities like walking.

Takeaway: While all models achieve comparable overall accuracy, TCN offers better handling of dynamic activities and slightly higher macro and weighted averages.

Summary and Conclusion

Model Comparison for Human Activity Recognition (HAR):

- **SVM:** Highest accuracy (96%) but struggles with sitting (88% recall).
- **CNN:** Efficient for short-term dependencies; accuracy: 92.16%, recall issues with sitting (85%) and standing (78%).
- **LSTM:** Captures long-term dependencies; accuracy: 92.37%, higher test loss than CNN.
- **CNN-LSTM:** Combines spatial and temporal modeling; accuracy: 91.82%.
- **TCN:** Best NN performance (93.01% accuracy); minor issues with static activities.

Key Takeaways:

- TCNs balance computational efficiency and accuracy, excelling in HAR tasks.
- Static activities like sitting and standing remain challenging and require refinement.

Future Work: Explore hybrid models integrating TCNs with other architectures to enhance static activity classification.