EMG2QWERTY: Exploring Deep Learning Architectures for sEMG-based Keystroke Prediction

Mehmet Yigit Turali (106530088) , Kartik Sharma (506530109)

Abstract

This project investigates various deep learning models for decoding electromyography (sEMG) signals into QWERTY keystrokes. We compare convolutional, recurrent, and transformer-based models. Our experiments reveal the effectiveness of hybrid architectures in reducing Character Error Rate (CER) and improving generalization across users.

1 Introduction

Interpreting surface electromyography (sEMG) signals for practical applications poses a complex problem in human-machine interaction, particularly for tasks like typing on a QWERTY keyboard. This project centers on enhancing the accuracy of converting muscle activity into precise keystrokes, a critical step for advancing assistive technologies and bio-signal interfaces. We delve into the potential of convolutional networks, recurrent networks, transformer-based architectures, and hybrid approaches to tackle this challenge. Our work focuses on surpassing the baseline Character Error Rate (CER) established by the TDS Conv model, driving the development of more accurate and effective sEMG-based keystroke prediction systems.

2 Methods

This section outlines the methodology, encompassing data preprocessing, model design, and evaluation strategies tailored to improve performance over baseline standards. The code can be accessed here [1].

2.1 Data Preprocessing

To ensure optimal model performance, raw sEMG signals undergo a series of preprocessing steps to enhance signal quality and consistency. These steps are critical for mitigating variability across recording sessions and users, laying a robust foundation for subsequent modeling efforts.

- **SincConv1D** – Substitutes traditional convolution with a parameterized sinc-based filter for more efficient and frequency-selective feature extraction from sEMG signals.
- **Electrode Rotation Augmentation** – Simulates variations in electrode placement to improve model robustness and adaptability across diverse user anatomies. This technique enhances cross-user generalization by mimicking real-world sensor misalignment.
- **Time Warping** – Applies controlled temporal distortions to the signal, enriching the training data and enhancing the model's ability to generalize across irregular input patterns. It effectively simulates natural variations in muscle activation timing.
- **Channel Attention** – This mechanism amplifies the focus on high-impact channels for improved prediction accuracy.

[1] https://github.com/YigitTurali/emg2qwerty.git

2.2 Model Architectures

We implement and compare multiple deep learning models:

- **TDS ConvNet (Base)** – A Time-Depth Separable CNN serving as the baseline model.
- **RNN** – A recurrent model that captures temporal dependencies in sEMG sequences.
- **CNN + GRU** – Uses CNN for feature extraction and GRU for efficient sequential modeling.
- **CNN + LSTM** – A hybrid approach leveraging CNNs for spatial features and LSTMs for temporal dependencies.
- **CNN + LSTM Alternating** – Alternates between CNN and LSTM layers for improved feature fusion.
- **CNN + Transformer** – Combines CNN feature extraction with Transformer-based attention mechanisms.

3 Results

Table 1 presents the test performance across different models.

Model	loss	CER (%)	IER (%)	SER (%)	DER (%)
TDS ConvNet (Base)	0.856	23.622	4.949	16.555	2.118
Baseline + preprocess	0.873	25.286	5.144	17.852	2.291
RNN	13.094	99.978	99.978	0.000	0.000
CNN + GRU	1.616	42.814	34.817	7.997	0.000
CNN + LSTM	**0.582**	**16.598**	**3.134**	12.016	1.448
CNN + LSTM Alternating	1.616	42.814	34.817	7.997	0.000
CNN + Transformer	3.091	66.047	58.353	7.435	0.259

Table 1: Performance comparison across different architectures, evaluating Loss, Character Error Rate (CER), Insertion Error Rate (IER), Sentence Error Rate (SER), and Deletion Error Rate (DER). All the models were trained for 150 epochs.

From the above table, it can be easily infered that CNN + LSTM performed the best so we tried to brute force it to even lower CER by training it for more epochs. Due to GPU constraints, the most we could do was 200 epochs and we got a CER of **15.776**.

4 Analysis

In this section, we conduct a comprehensive and detailed analysis of the various deep learning architectures implemented for the sEMG-based keystroke prediction task. We systematically evaluate their effectiveness by exploring theoretical foundations, empirical performance, and loss and Character Error Rate (CER) trends.

4.1 Performance Comparison: RNN vs. CNN-Based Models

Recurrent Neural Networks (RNNs) are conventionally used for sequential data modeling; however, they demonstrate several critical limitations when applied to sEMG-based keystroke decoding:

- **Vanishing Gradient Problem:** RNNs suffer from vanishing gradients, which restricts their ability to learn long-range dependencies. This results in a loss of critical information from earlier time steps, especially in long sEMG sequences.
- **Lack of Spatial Feature Extraction:** sEMG signals contain structured spatial dependencies across multiple electrode channels. CNNs excel at capturing these local features, while RNNs process signals as flat sequences, leading to suboptimal spatial encoding.

Based on these observations, we conclude that CNNs are indispensable for effective feature extraction, necessitating their inclusion in all model architectures.

4.2 CNN+LSTM Alternating vs. Standard CNN+LSTM

We investigated an alternative design where CNN and LSTM layers alternate instead of using a sequential pipeline. However, this approach performed worse than the standard CNN+LSTM. We believe that it could be due to the following reasons:

- **Disruptive Feature Hierarchy:** Alternating between CNN and LSTM layers leads to inconsistent feature representations, preventing the network from forming a structured spatial-temporal hierarchy.
- **Increased Computational Complexity:** The alternating architecture introduces additional parameter overhead, increasing computational cost without substantial performance improvements.
- **Training Instability:** The training loss plots indicate frequent oscillations, suggesting difficulties in learning consistent representations across layers.

Thus, our empirical results advocate for a structured CNN+LSTM pipeline instead of an alternating configuration.

4.3 Why attention-based models don't help (Attention isn't all you need?)

Transformers and attention-based architectures have demonstrated remarkable success in NLP and other sequence-based tasks. However, they were not beneficial in this specific domain due to:

- **Mismatch in Temporal Requirements:** Transformers are designed for capturing long-range dependencies efficiently, but sEMG signals primarily rely on short-range interactions, which LSTMs handle more effectively.
- **Lack of Strong Positional Encoding:** Unlike text sequences, sEMG signals do not have strong token-based structures where self-attention mechanisms thrive. Consequently, transformers fail to leverage their strengths effectively.

Our findings indicate that while transformers offer flexibility, they do not provide sufficient advantages in sEMG-based keystroke prediction.

4.4 Loss and CER Analysis

The training and validation processes for each model reveal significant insights into their strengths, weaknesses, and ability to generalize.

Figure 1 provides a comparative analysis of training and validation loss trends across all different models. Figure 2 presents the Character Error Rate (CER) for training and validation phases across different models. A key observation is that validation CER for RNN and CNN+Transformer remains high, indicating poor generalization and alignment issues in CTC decoding. CNN+LSTM and CNN+GRU show significantly lower CER values, affirming their robustness in handling sEMG-based keystroke prediction.

Figure 3 compares CNN+LSTM trained for 150 and 200 epochs to examine whether additional training epochs lead to performance improvements. The loss curves for both training and validation show minimal differences between 150 and 200 epochs, indicating that training beyond 150 epochs does not significantly improve model performance. Additionally, validation CER stagnates at 100, suggesting that the primary limitation is in the CTC decoding stage rather than the number of training iterations.

4.4.1 CNN+LSTM Training and Validation

As shown in Figure 1, the CNN+LSTM model exhibits a smooth and steady decrease in training loss, demonstrating effective learning. Initially, training loss starts at approximately 5.95 and gradually decreases, stabilizing between 3.3 and 3.5. However, a notable concern is the plateauing of

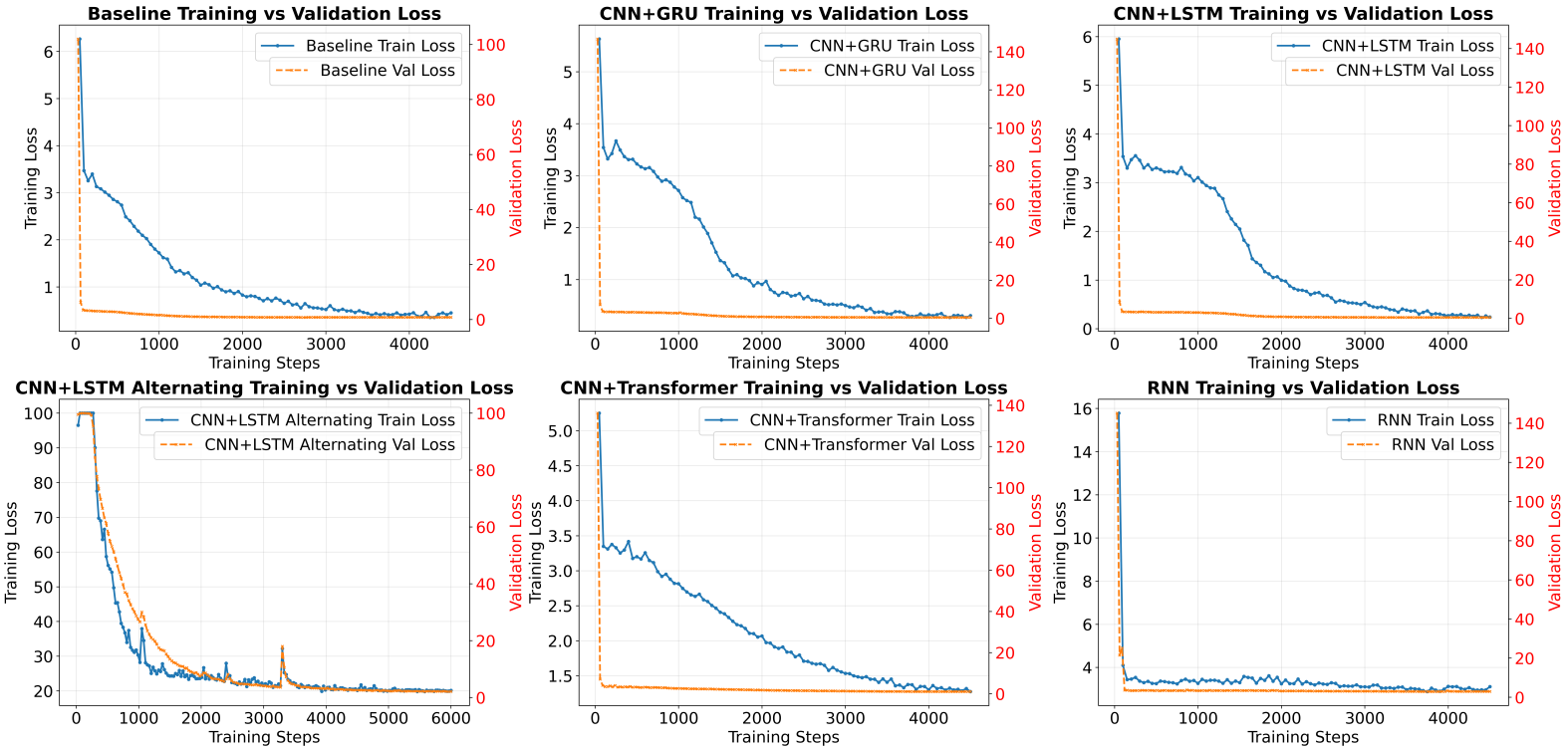

Figure 1: Training vs. Validation Loss Across Models

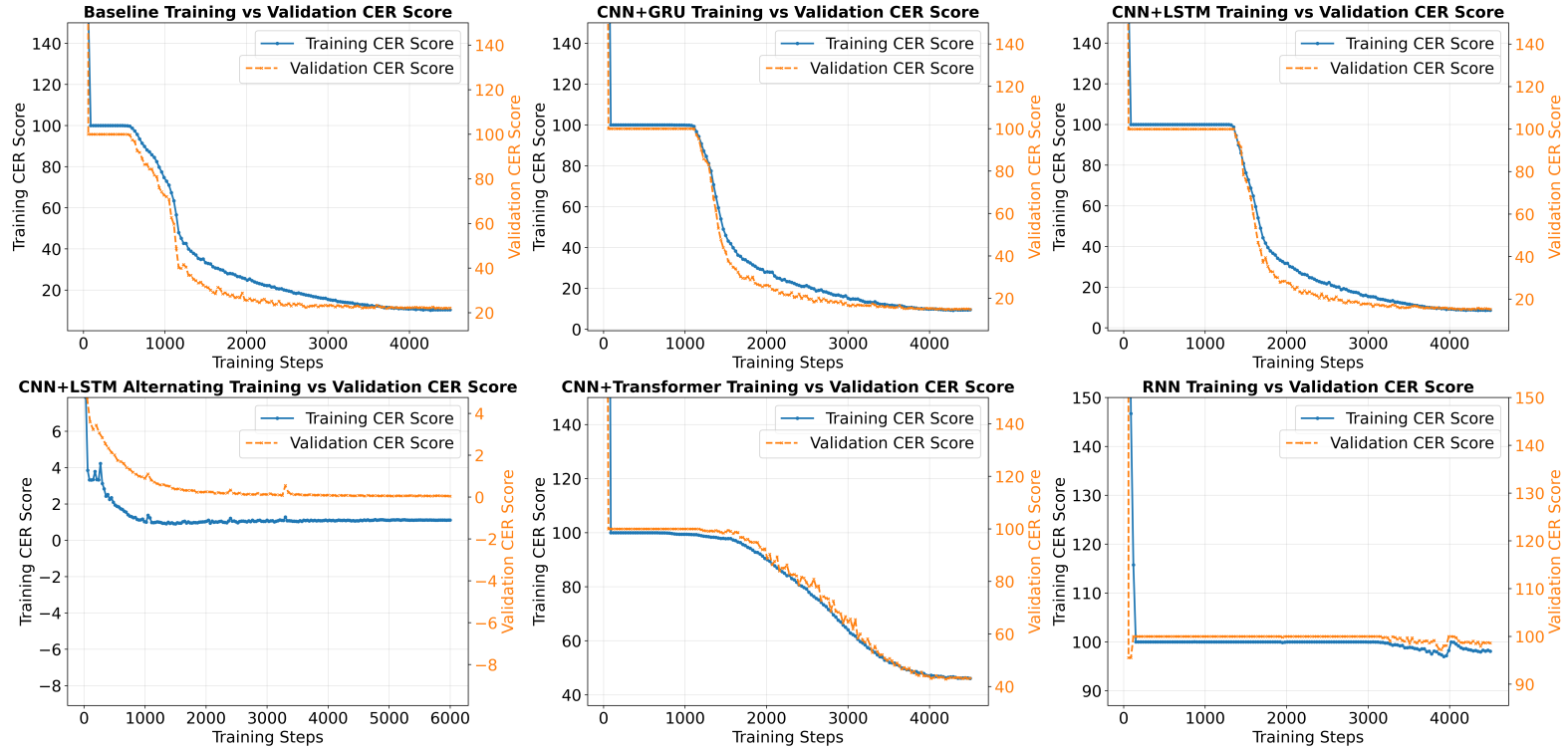

Figure 2: Training vs. Validation CER Across Models

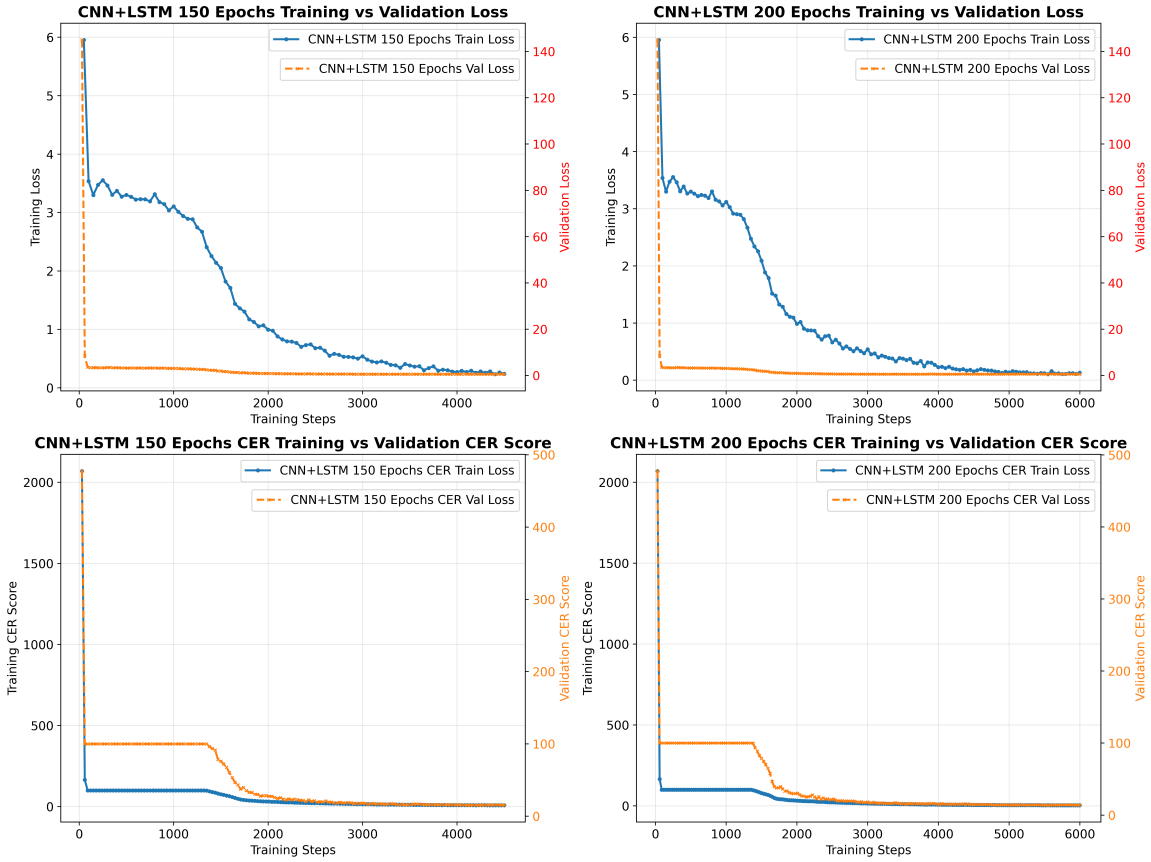

Figure 3: CNN+LSTM 150 vs. 200 Epochs Training and Validation Loss

validation CER at 100 (Figure 2), suggesting alignment issues in the decoding phase. Despite this, CNN+LSTM remains the best-performing model in terms of training stability, loss convergence, and generalization capability.

4.4.2 CNN+GRU Training and Validation

The CNN+GRU model exhibits a different pattern in training loss, as illustrated in Figure 1. While training loss starts at a comparable level to CNN+LSTM, the initial fluctuations are more pronounced, suggesting instability in gradient updates. This aligns with the characteristic behavior of GRUs, which have fewer parameters than LSTMs but may struggle with capturing complex dependencies. The validation loss (Figure 1) fluctuates significantly more than CNN+LSTM, implying difficulties in generalizing to unseen data. The validation CER, shown in Figure 2, also suffers from the same plateauing issue at 100, reinforcing that CNN+GRU, while computationally efficient, does not outperform CNN+LSTM in robustness and stability.

4.4.3 CNN+LSTM Alternating Training and Validation

The alternating CNN+LSTM model, which interleaves CNN and LSTM layers instead of stacking them sequentially, was tested to explore potential benefits in hierarchical feature extraction. However, as Figure 1 shows, the loss progression is more erratic than the standard CNN+LSTM model. The frequent layer switching appears to disrupt the consistency of learning, leading to less stable optimization. The validation loss, illustrated in Figure 1, further highlights this instability, as it does not stabilize as smoothly as in other models. Additionally, the validation CER trend in Figure 2 follows the same plateauing behavior, reinforcing that the alternating architecture does not provide an advantage over the standard CNN+LSTM configuration.

4.4.4 RNN Training and Validation

The RNN model, which lacks convolutional layers, struggles significantly in both training and validation phases. Figure 1 shows that the initial training loss starts much higher at approximately 15.78, indicating inefficient learning. Although the loss eventually drops, it does not reach the same stable levels observed in CNN-based models. The validation loss, as seen in Figure 1, exhibits extreme fluctuations, suggesting poor generalization. The validation CER in Figure 2 follows an even worse trend, with no significant reduction and a prolonged plateau at 100. This confirms that standard RNNs are inadequate for this task due to their inability to process spatial dependencies effectively.

4.4.5 CNN+Transformer Training and Validation

The CNN+Transformer model was evaluated to determine whether self-attention mechanisms could enhance performance. As seen in Figure 1, the training loss starts lower than RNNs but remains higher than CNN+LSTM throughout training. The validation loss (Figure 1) remains unstable, indicating that the model struggles to generalize. A key limitation of the Transformer architecture is its reliance on long-range dependencies, which are less relevant in sEMG-based keystroke prediction. This is reflected in the validation CER trend in Figure 2, where the model reaches a plateau at 100, similar to other underperforming models. This confirms that Transformers are not well-suited for this specific task due to their inefficient handling of short-term dependencies.

4.4.6 Extended Training Analysis (150 vs. 200 Epochs)

To assess whether extending training beyond 150 epochs provides additional benefits, we compared CNN+LSTM models trained for 150 and 200 epochs. As illustrated in Figure 3, there is no meaningful improvement in validation loss beyond 150 epochs. Training loss continues to decrease slightly, but validation loss and CER remain stagnant. This suggests that additional training does not translate to better generalization, reinforcing the need for early stopping mechanisms to prevent overfitting.

The combined findings from these training and validation analyses highlight the importance of robust model selection, regularization techniques, and improved decoding strategies. CNN+LSTM remains the most reliable model, but addressing persistent validation CER plateauing through better data augmentation and decoding refinements will be crucial for further improvements in sEMG-based keystroke prediction.

5 Architecture of the best model

Here, in Figure 4 we provide a schematic architecture of the model which gave the best CER, i.e., CNN-LSTM.

6 Further Improvements

While the CNN+LSTM model outperformed others, our preprocessing techniques—SincConv1D, Electrode Rotation Augmentation, and Time Warping—did not improve performance as expected. SincConv1D's frequency selectivity, electrode rotation's variability simulation, and time warping's temporal distortions may have misaligned with sEMG signal needs, suggesting a need for refined or alternative preprocessing strategies.

We briefly explored wavelet transforms to capture time-frequency features but couldn't optimize them due to time constraints. Preliminary results were promising, and further investigation into wavelet families and decomposition levels could enhance signal representation. Time and GPU limitations also capped training at 200 epochs, restricting deeper exploration of architectures, augmentations like GANs and mainly VAEs.

7 Conclusion

Our investigation demonstrates that the CNN-LSTM hybrid model outperforms other architectures, surpassing the baseline by a substantial margin. Contrary to expectations, transformer-based models, while capable of incremental accuracy gains, did not prove superior in this context and de-

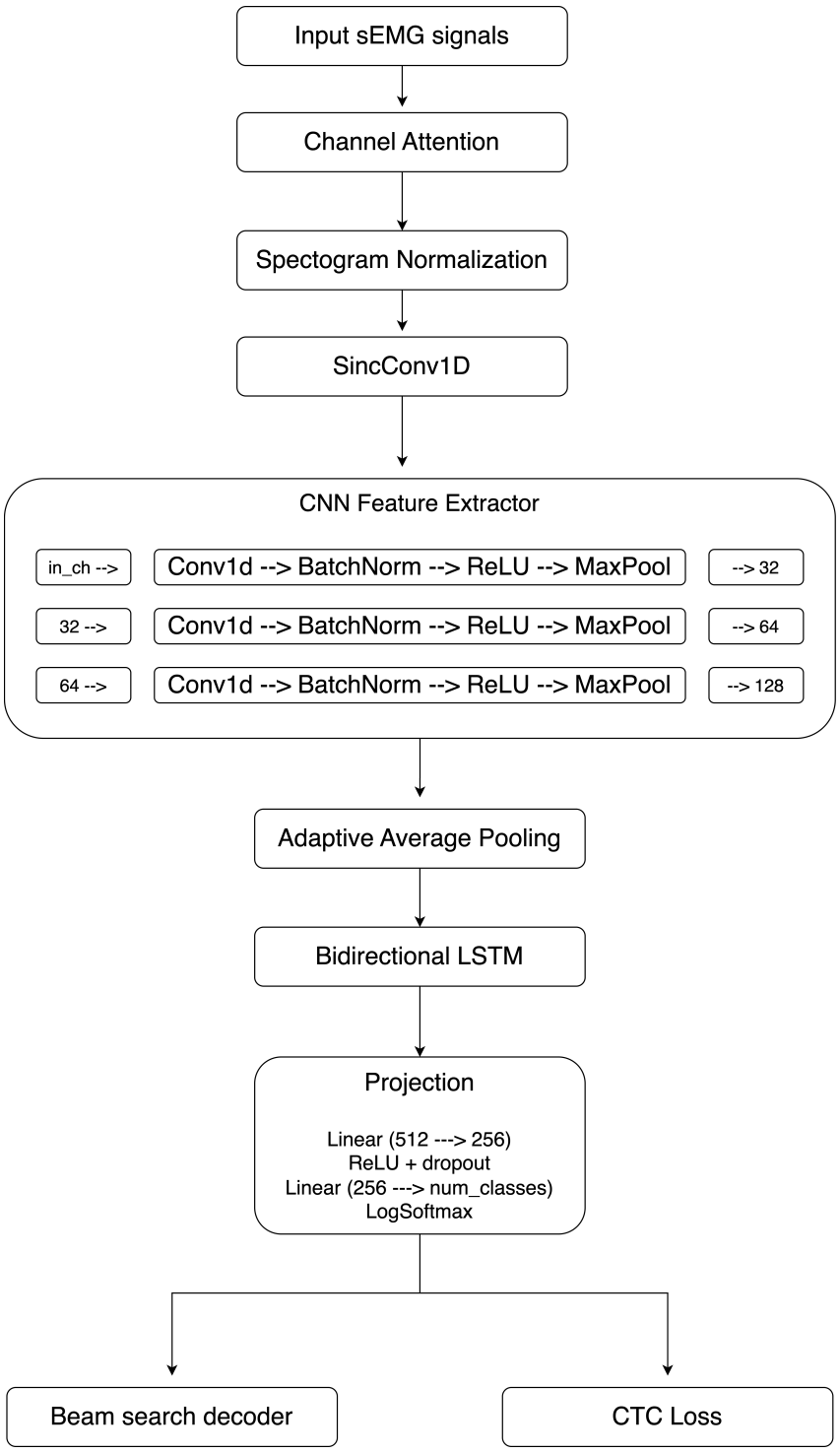

Figure 4: Architechture of the CNN-LSTM model (best CER).

mand significantly larger datasets for optimal performance. Our findings underscore the efficacy of hybrid architectures, which consistently achieve lower Character Error Rates (CER) compared to standalone CNN or RNN models. These results highlight the potential of integrating convolutional and recurrent layers to capture both spatial and temporal dependencies in sEMG signals, offering a robust solution for keystroke prediction in real-world applications.

References

[1] J. Lin et al., "EMG2QWERTY: Decoding Keystrokes from Surface EMG Signals," arXiv:2410.20081, 2024.

[2] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is All You Need," in *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 30, 2017.

[3] J. Lin et al., "EMG2QWERTY: Decoding Keystrokes from Surface EMG Signals," arXiv:2410.20081, 2024.

[4] J. Kim et al., "AI-Based Stroke Disease Prediction System Using Real-Time Electromyography Signals," *Applied Sciences*, 2023.

[5] M. C. Mora, J. V. García-Ortiz, and J. Cerdá-Boluda, "sEMG-Based Robust Recognition of Grasping Postures with a Machine Learning Approach for Low-Cost Hand Control," *Sensors*, vol. 24, no. 6, 2063, 2024.

[6] A. Sultana, F. Ahmed, and M. S. Alam, "A Systematic Review on Surface Electromyography-Based Classification System for Identifying Hand and Finger Movements," *Healthcare Analytics*, vol. 3, 100126, 2023.

[7] H. Zhang et al., "Domain Contrast Network for Cross-Muscle ALS Disease Identification with EMG Signal," *Biomedical Signal Processing and Control*, vol. 82, 104582, 2023.

[8] E. Pérez-Giraldo et al., "Multimodal Deep Learning Model for Cylindrical Grasp Prediction Using Surface Electromyography and Contextual Data During Reaching," *Sensors*, vol. 25, no. 4, 2025.

[9] Z. Wei, M. Li, Z.-Q. Zhang, and S. Xie, "Continuous Prediction of Wrist Joint Kinematics Using Surface Electromyography from the Perspective of Muscle Anatomy and Muscle Synergy Feature Extraction," *Journal of Biomedical and Health Informatics*, 2025.