



An ensemble of long short-term memory networks with an attention mechanism for upper limb electromyography signal classification

LSE Research Online URL for this paper: <http://eprints.lse.ac.uk/123550/>

Version: Published Version

Article:

Alotaibi, Naif D., Jahanshahi, Hadi, Yao, Qijia, Mou, Jun and Bekiros, Stelios (2023) An ensemble of long short-term memory networks with an attention mechanism for upper limb electromyography signal classification. *Mathematics*, 11 (18). ISSN 2227-7390

<https://doi.org/10.3390/math11184004>

Reuse

This article is distributed under the terms of the Creative Commons Attribution (CC BY) licence. This licence allows you to distribute, remix, tweak, and build upon the work, even commercially, as long as you credit the authors for the original work. More information and the full terms of the licence here: <https://creativecommons.org/licenses/>

Article

An Ensemble of Long Short-Term Memory Networks with an Attention Mechanism for Upper Limb Electromyography Signal Classification

Naif D. Alotaibi ¹, Hadi Jahanshahi ^{2,*} , Qijia Yao ³ , Jun Mou ⁴  and Stelios Bekiros ^{5,6,7,*} 

¹ Communication Systems and Networks Research Group, Department of Electrical and Computer Engineering, Faculty of Engineering, King Abdulaziz University, Jeddah 21589, Saudi Arabia; ndalotabi@kau.edu.sa

² Institute of Electrical and Electronics Engineers, Toronto, ON M5V 3T9, Canada

³ School of Automation and Electrical Engineering, University of Science and Technology Beijing, Beijing 100083, China; qijia_yao@ustb.edu.cn

⁴ School of Information Science and Engineering, Dalian Polytechnic University, Dalian 116034, China; moujun@csu.edu.cn

⁵ FEMA, University of Malta, MSD 2080 Msida, Malta

⁶ LSE Health, Department of Health Policy, London School of Economics and Political Science, London WC2A 2AE, UK

⁷ IPAG Business School, 184 Boulevard Saint-Germain, 75006 Paris, France

* Correspondence: jahanshahi.hadi@ieee.org (H.J.); stelios.bekiros@um.edu.mt (S.B.)

Abstract: Advancing cutting-edge techniques to accurately classify electromyography (EMG) signals are of paramount importance given their extensive implications and uses. While recent studies in the literature present promising findings, a significant potential still exists for substantial enhancement. Motivated by this need, our current paper introduces a novel ensemble neural network approach for time series classification, specifically focusing on the classification of upper limb EMG signals. Our proposed technique integrates long short-term memory networks (LSTM) and attention mechanisms, leveraging their capabilities to achieve accurate classification. We provide a thorough explanation of the architecture and methodology, considering the unique characteristics and challenges posed by EMG signals. Furthermore, we outline the preprocessing steps employed to transform raw EMG signals into a suitable format for classification. To evaluate the effectiveness of our proposed technique, we compare its performance with a baseline LSTM classifier. The obtained numerical results demonstrate the superiority of our method. Remarkably, the method we propose attains an average accuracy of 91.5%, with all motion classifications surpassing the 90% threshold.

Keywords: EMG signals; time series classification; neural network; LSTM; attention mechanism

MSC: 68T05; 92B20; 68T45; 68U10



Citation: Alotaibi, N.D.; Jahanshahi, H.; Yao, Q.; Mou, J.; Bekiros, S. An Ensemble of Long Short-Term Memory Networks with an Attention Mechanism for Upper Limb Electromyography Signal Classification. *Mathematics* **2023**, *11*, 4004. <https://doi.org/10.3390/math11184004>

Academic Editors: Paolo Crippa and Paolo Mercorelli

Received: 22 July 2023

Revised: 5 September 2023

Accepted: 19 September 2023

Published: 21 September 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

EMG signals, which are electrical impulses produced by muscle activities, hold significant value across various fields due to their broad scope of application [1–3]. Their pivotal role within the healthcare arena is particularly noteworthy. Interpreted as a distinctive bio-electrical language employed by muscles, EMG signals equip medical professionals with a powerful tool to assess muscular function, identify anomalies, and diagnose pathologies pertaining to the nervous system [4,5]. Rigorous analysis of these signals yields a comprehensive depiction of the interrelated health of the muscular and neurological systems. This affords clinicians the ability to navigate toward optimal treatment modalities and interventions for a myriad of neuromuscular disorders [6].

While the foundational understanding of EMG signals is well-established, there are still gaps in effectively leveraging this knowledge, especially in the realm of myoelectric

control. Current approaches often falter in real-time applications, especially when signal noise and environmental factors come into play. Our research, therefore, aims to address these challenges, shedding light on the intricate problem of myoelectric control and its applications.

Beyond medicine, EMG signals have found practical applications in prosthetics and assistive technologies. By capturing and interpreting EMG signals, researchers and engineers have developed advanced prosthetic limbs that can be controlled by the user's muscle contractions, restoring mobility and independence to individuals with limb loss [7–9]. This technology has revolutionized the lives of many, offering precise and intuitive control over prosthetic devices [10,11]. Moreover, EMG signals play a crucial role in sports science and rehabilitation. By monitoring muscle activation patterns through EMG recordings, researchers and therapists can analyze movement patterns, design tailored rehabilitation programs, optimize training regimes, and prevent injuries [12]. EMG-based biofeedback systems provide real-time information, allowing individuals to adjust their movements and enhance their performance [13]. Overall, EMG signals have proven to be a valuable tool in understanding muscle function, diagnosing disorders, improving prosthetic technology, and optimizing rehabilitation and sports performance.

The classification of EMG signals holds paramount importance due to its wide-ranging applications and implications across various fields. Accurate and reliable classification of EMG signals enables the precise assessment and understanding of muscle activity, aiding in the diagnosis and monitoring of neuromuscular disorders [14–16]. This classification is instrumental in the development of assistive technologies, such as prosthetics, allowing for the intuitive control and restoration of mobility for individuals with limb loss. Additionally, EMG signal classification plays a vital role in rehabilitation and physical therapy, facilitating personalized treatment plans based on each patient's unique muscular responses [17]. It also contributes to sports science and biomechanics by analyzing muscle activation patterns, enhancing athletic performance, and preventing injuries [18,19].

Neural networks offer significant benefits for the classification of time series signals [20–22]. The inherent ability of neural networks to learn complex patterns and relationships within data makes them well-suited for capturing the temporal dependencies present in time series signals. Long Short-Term Memory (LSTM), a specialized type of neural network, excels in modeling sequential data by effectively retaining and utilizing historical context through their memory cells [23,24]. This makes them particularly effective in handling long-term dependencies and capturing temporal dynamics. In the context of signal classification, neural networks and LSTMs can extract meaningful features from time series signals, automatically learn discriminative patterns, and make accurate predictions. Their ability to handle large amounts of data and adapt to different signal characteristics makes them invaluable for tasks such as speech recognition, gesture recognition, medical signal analysis, and financial market forecasting [25,26].

Many approaches have been proposed for EMG signal classification, and varying levels of accuracy have been obtained [27]. To name a few, in [28], an accuracy range of 75–84% is reported. Similarly, in [29], an accuracy of 77.85% is documented. For a comprehensive analysis, readers are referred to [30]. However, when comparing these results, it is essential to note that different datasets were used, and they possess varying levels of accuracy.

In [31], the authors designed an HMI integrating EEG and sEMG signals for controlling lower extremity exoskeletons. Through a deep analysis of human walking, they achieved a high level of accuracy. However, this being a preliminary study, broader validation is needed. The simultaneous need for both EEG and sEMG signals might also pose challenges in practical applications. In [32], a novel method was proposed that combines multiple signals from the lower extremity. By simulating human learning processes, they managed to reconstruct these signals to estimate joint angles. Their approach has outperformed other techniques in terms of the error rate. However, the limited number of participants (eight) could raise concerns about the generalizability of the results, and the complexity of multiple

signal streams might challenge real-world applications. In [33], the potential of transient sEMG signals for myoelectric control was investigated with both intact-limb subjects and transhumeral amputees. The 100 ms window length has emerged as optimal for classification. By focusing on transhumeral amputees, the study addressed a significant yet niche group. However, its niche focus could limit its broader applicability. In essence, these studies have highlighted the advancements and challenges in human–machine interfaces tailored for exoskeletons and assistive devices.

Given its wide-ranging implications and applications, there is still a pressing need to develop state-of-the-art classification techniques for accurately classifying EMG signals [34,35]. Accurate classification enables healthcare professionals to precisely diagnose and monitor neuromuscular disorders, allowing for timely intervention and tailored treatment plans. It facilitates the development of advanced prosthetics and assistive technologies that can be controlled intuitively by EMG signals, enhancing mobility and independence for individuals with limb loss. Furthermore, state-of-the-art classification techniques contribute to the optimization of rehabilitation programs by accurately assessing muscle activation patterns and designing personalized treatment strategies. Consequently, the development of state-of-the-art classification techniques for EMG signals empowers healthcare, rehabilitation, and assistive technology fields, leading to improved patient outcomes, enhanced quality of life, and advancements in human performance.

We propose an ensemble classification method for EMG signals, which offers several significant contributions. Firstly, our approach introduces a novel ensemble architecture specifically tailored for time series classification. This architecture combines projected LSTM models, harnessing their complementary strengths to enhance classification accuracy. Secondly, our method demonstrates efficiency by achieving fast processing speeds, enabling real-time or near-real-time applications. Thirdly, the ensemble model effectively addresses the challenge of overfitting by leveraging the diversity among individual models, resulting in improved robustness and generalization. Additionally, our proposed method exhibits resilience against noise, making it suitable for real-world EMG signal classification scenarios. Overall, the contributions of this study encompass enhanced accuracy, speed, noise robustness, and prevention of overfitting through the novel ensemble architecture for EMG signal classification.

The structure of the remainder of this study unfolds as follows: Section 2 provides an overview of RNN, delves into LSTM neural networks, and explores the advantages of Attention LSTM (ALSTM). Following this, in Section 3, we introduce and elaborate on our proposed ensemble technique. Section 4 details the handling of EMG time-series data and the preprocessing stages required for classification. In Section 5, we apply our proposed ensemble ALSTM and compare its performance with the conventional LSTM. In Section 6, we provide an in-depth analysis of the results obtained. Finally, Section 7 furnishes concluding remarks and future research recommendations.

2. RNN for Classification

RNN serves as a powerful tool for classification tasks due to its unique ability to process sequential data, making it particularly effective in situations where the temporal dynamics of the input carry crucial information [36,37]. RNNs, with their inherent ability to process and remember sequential data, are particularly beneficial for time series classifications. As a result, RNNs have been increasingly adopted for EMG signal classification to capture the temporal dynamics of muscle activations [38,39]. Unlike traditional feedforward networks, RNNs are designed with feedback connections that allow information to cycle through the network over time, providing the capacity for memory of past inputs [40,41]. This characteristic allows RNNs to exploit temporal dependencies, making them adept at tasks such as speech recognition, sentiment analysis, and time-series prediction. By training an RNN on the appropriate labeled data, we can classify new unseen sequences based on learned patterns, demonstrating its utility in diverse classification contexts.

2.1. LSTM

LSTM is a type of RNN architecture that has gained significant popularity in the field of time series classification. Unlike traditional RNNs, LSTM networks are designed to effectively capture long-term dependencies and temporal dynamics in sequential data [42]. This is achieved through the use of memory cells, which can retain and update information over extended sequences. The gating mechanisms within LSTM allow for the selective retention and utilization of relevant contextual information, making it highly suitable for modeling complex patterns in time series data.

LSTM networks offer several advantages for time series classification tasks [43]. Firstly, they can handle long-term dependencies, which is crucial for accurately capturing temporal patterns that may span across extended periods. This makes LSTM particularly effective in scenarios where the context and history of the data play a significant role. Secondly, LSTM networks are capable of learning and retaining information over time, allowing them to effectively model and adapt to changing patterns in the time series data. This adaptability is essential in dynamic environments where the characteristics of the data may vary. Lastly, LSTM networks are able to automatically extract relevant features from the input time series, alleviating the need for manual feature engineering and reducing the risk of missing important information.

The application of LSTM networks in the classification of EMG signals brings several advantages. LSTM's ability to capture temporal dependencies makes it well-suited for this task. LSTM networks can effectively model the complex dynamics of muscle contractions and relaxations, enabling accurate classification of EMG signals. The adaptability of LSTM allows it to handle variations in signal characteristics and capture subtle changes in muscle activation patterns over time. This makes LSTM a valuable tool for diagnosing neuromuscular disorders, designing prosthetics, and optimizing rehabilitation programs in the field of EMG signal analysis.

Figure 1 shows an LSTM cell. As depicted in this figure, an incoming series of data points represented as $i = [i^1, i^2, \dots, i^T]$ is fed into the network. This then results in a hidden sequence denoted by $h = [h^1, h^2, \dots, h^T]$. This transformation occurs as each network activation unit processes the sequence in an iterative manner over time, adhering to the following equations:

$$g_t = \sigma(W_{gi}i_t + W_{gh}h_{t-1} + b_g) \quad (1)$$

$$m_t = \sigma(W_{mi}i_t + W_{mh}h_{t-1} + b_m) \quad (2)$$

$$\tilde{j}_t = \tanh(W_{ji}i_t + W_{jh}h_{t-1} + b_j) \quad (3)$$

$$n_t = \sigma(W_{ni}i_t + W_{nh}h_{t-1} + b_n) \quad (4)$$

$$j_t = (g_t \odot j_{t-1}) + (m_t \odot \tilde{j}_t) \quad (5)$$

$$h_t = n_t \odot \tanh(j_t) \quad (6)$$

In this context, h_t refers to the present time step's hidden state output. Also, the forget, input, and output gates are represented by g , m , and n , correspondingly. The candidate cell vector is illustrated by \tilde{j}_t , while j symbolizes the cell activation vector. The weight matrices and bias vector are designated by the W and b terms, respectively. The symbol \odot is used to signify the Hadamard product. The activation functions for sigmoid and hyperbolic tangent are expressed through the use of σ and \tanh , respectively.

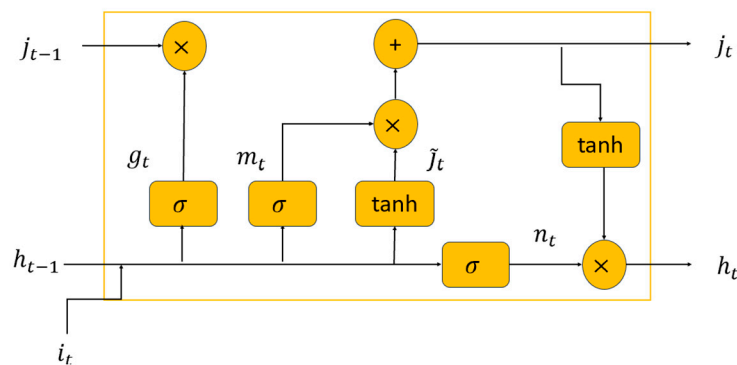


Figure 1. Long short-term memory cell.

As shown in the abovementioned formulations and Figure 1, within the architecture of an LSTM cell, there exist multiple gates, and each is necessitated to employ a specific activation function. The Sigmoid (or Logistic) function is one such function, proffering outputs that range between 0 and 1. It is predominantly utilized for the forget, input, and output gates within an LSTM cell, facilitating meticulous control over the dissemination of information. Another prevalent activation function is the Hyperbolic Tangent (tanh) function, producing outputs that span between -1 and 1 . Within the context of LSTMs, the tanh function is commonly employed for the input modulation gate, governing the potential values that may be incorporated into the cell state. A less conventional function used in certain neural network paradigms, though less so in LSTMs, is the Rectified Linear Unit (ReLU). This function relays the input directly if it is positive, and otherwise, it yields zero. Its computational expediency is noteworthy. Additionally, there is the Leaky ReLU, a derivative of the ReLU, which permits a minimal gradient even in the absence of unit activation, thereby facilitating the preservation of gradient flow during the backpropagation process (see reference [44] for detailed information on the activations functions for LSTM).

In our study, we have elected to utilize the tanh function as our primary activation function, attributing to its superior properties. The tanh function’s zero-centered output can expedite convergence, and its adeptness in processing negative values is of significance. Its inherent squashing mechanism, which maps values between -1 and 1 , is instrumental in curtailing the challenges posed by the exploding gradient phenomenon. Furthermore, the uninterrupted gradient of the tanh function ensures that the LSTM modules sustain their learning trajectory across elongated sequences, thereby adeptly capturing long-term dependencies.

2.2. ALSTM

ALSTM is a potent combination of two crucial concepts in the realm of deep learning and natural language processing. The addition of the attention mechanism allows the model to dynamically focus on different parts of the input sequence for making predictions, essentially learning what to “pay attention to” based on the task at hand [45,46]. This imbues the model with the ability to create context-sensitive representations, providing significant improvements in performance across many tasks, including machine translation, speech recognition, and time series analyses.

As portrayed in Figure 2, the ALSTM cell generates two distinct outputs: the cell state j_t , which serves as the repository for long-term memory, and the hidden state h_t , that handles short-term memory storage. In this study, we incorporated a projected layer following the output gate n_t to shrink the size of the hidden state output. Suppose ‘p’ and ‘m’ represent the projected and hidden sizes of the ALSTM, respectively. The output size of the hidden state is then represented as h_t in the space of $\mathbb{R}^{p \times 1}$, and the cell state is expressed as j_t in the space of $\mathbb{R}^{m \times 1}$. At each time step t , the ALSTM cell accepts a vector corresponding to an input feature. The hidden and cell states from the prior time step ($t - 1$) are used as additional inputs. In this proposed architecture, the key components,

which are the four gates—the forget gate g_t , the input gate m_t , the candidate cell gate $\tilde{j}_t \in \mathbb{R}^{m \times 1}$, and the output gate $n_t \in \mathbb{R}^{m \times 1}$, operate in a manner identical to the traditional LSTM.

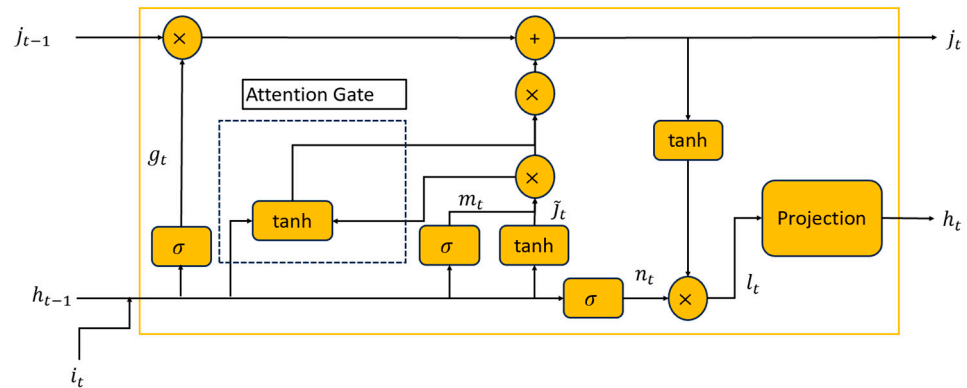


Figure 2. The structure of an LSTM cell with an attention mechanism.

In addition to the input gate, m_t governs the information flow into the memory cell, while the forget gate g_t determines how this information is updated within the memory cell. The output gate n_t then controls the information that will be carried forward to the next time step. The output information from the input gate m_t and the candidate cell gate \tilde{j}_t often lack the necessary focus, which could result in missing out on key information related to the cell gate memory j_t . To overcome this, we introduce an attention gate designed to hold onto important information for the cell gate memory. This gate aims to discover the correlation between the previous hidden state output h_{t-1} and the Hadamard product of the input gate m_t and candidate cell gate \tilde{j}_t . The mathematical representation of the proposed attention gate is given below:

$$\varphi_t = R_\varphi^T \tanh(W_\varphi(m_t \odot \tilde{j}_t) + Y_\varphi h_{t-1}) \tag{7}$$

$$\hat{j}_t = \text{softmax}(\varphi_t) \odot W_j(m_t \odot \tilde{j}_t). \tag{8}$$

In Equations (7) and (8), R_φ^T , W_s , Y_φ , and W_C represent trainable parameters, with φ_t acting as the attention-scoring function. The transformation formulas defined for the proposed ALSTM model are provided below:

$$g_t = \sigma(W_g i_t + U_g h_{t-1} + b_g) \tag{9}$$

$$m_t = \sigma(W_m i_t + U_m h_{t-1} + b_m) \tag{10}$$

$$\tilde{a}_t = \tanh(W_{\tilde{a}} i_t + U_{\tilde{a}} h_{t-1} + b_a) \tag{11}$$

$$j_t = (g_t \odot j_{t-1}) + \hat{j}_t \tag{12}$$

$$l_t = n_t \odot \tanh(j_t) \tag{13}$$

$$h_t = W_h l_t. \tag{14}$$

As per the given formulas, the input gate m_t dictates the nature of the information that will enter the cell state. The attention gate is utilized to process the transformations in Equations (7) and (8). This approach significantly enhances the predictive capabilities

of the traditional LSTM, and it achieves high accuracy when handling complex datasets characterized by noise and a broader spectrum of operational conditions.

3. Ensemble Architecture and Proposed Ensemble Technique

In this section, we first delve into the concept of ensemble neural networks. Subsequently, we introduce our proposed ensemble architecture. Lastly, we highlight the advantages of this ensemble approach, particularly for EMG signal classification.

3.1. Ensemble Architecture for Classification

An ensemble method in machine learning involves the combination of several models to solve a single prediction problem [47]. It operates by constructing a set of classifiers and then classifying new data points by taking a (weighted) vote on their predictions. The underlying principle behind ensemble methods is that different algorithms may have varied misclassification rates in different regions of the input space. By strategically combining these models, we aim to produce a more robust model with better generalization performance.

3.2. Proposed Ensemble Technique

Ensemble architecture, involving the orchestration of multiple neural networks, is pivotal in enhancing the effectiveness of classification systems, with its application playing a paramount role in a diverse range of classification tasks. This architecture amalgamates predictions from numerous individual models—commonly known as base learners or weak classifiers—which are trained independently. The final classification verdict results from this collective decision-making process.

In the method we propose, the outputs of individual classifiers undergo one-hot encoding, are subsequently integrated with the raw time-series signal, and are finally utilized in the ultimate classifier, which is the LSTM with an attention mechanism. Figure 3 visually presents the structure of the proposed classifier for EMG signal classification. The core philosophy behind our ensemble architecture is that the innate diversity among the individual models contributes to a substantial improvement in the system's accuracy, robustness, and generalization.

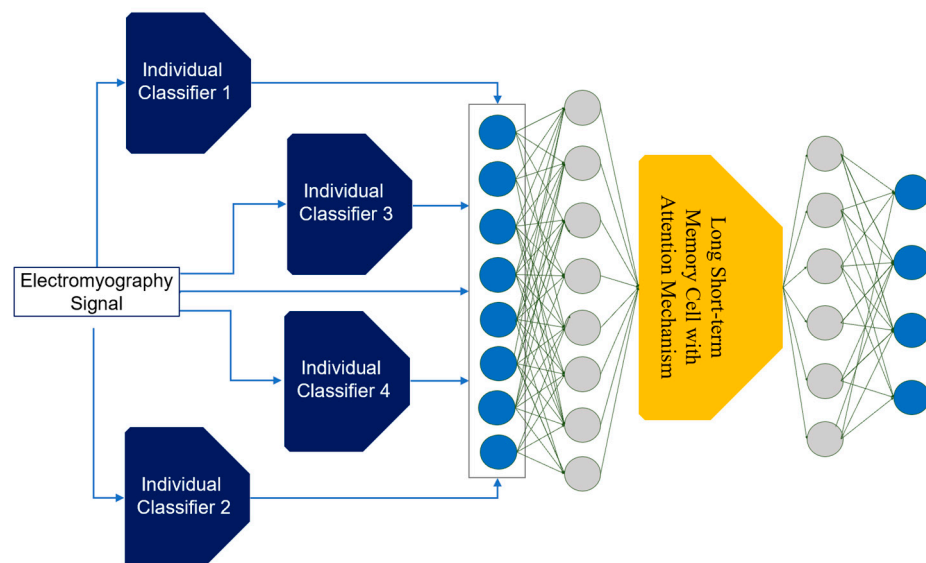


Figure 3. Architecture of the proposed classifier for EMG signal classification.

It should be highlighted that within our established framework, we have incorporated four individual classifiers. Although the architecture permits the integration of a greater number of classifiers, it is imperative to weigh the computational implications against the potential enhancements in classification accuracy when considering such augmentations.

3.3. Benefits of the Proposed Approach for EMG Classification

The application of the proposed ensemble architecture in time series classification, specifically in EMG signal analysis, brings specific advantages. EMG signals are time-varying and complex, representing muscle activation patterns. Ensemble architecture helps address the challenges associated with the inherent variability and noise in EMG signals. By combining the predictions of multiple models, the ensemble model can capture a broader range of patterns and better adapt to the diverse characteristics of EMG signals. This enhances the accuracy of EMG signal classification, aiding in the diagnosis of neuromuscular disorders, optimizing prosthetic control, and improving rehabilitation outcomes. Moreover, ensemble architecture provides robustness against noisy EMG signals, enabling reliable analysis and reducing the impact of signal artifacts on the classification results.

Here we list the key advantages that emerge from using an ensemble approach for EMG signal classification:

(1) Diversity of Model Predictions: EMG signals are complex time-series data that can vary greatly across different muscles, individuals, and activities. No single model might be optimal for all scenarios. An ensemble approach allows us to harness the strengths of diverse models, ensuring a more comprehensive understanding of the signals.

(2) Robustness to Noise: EMG signals are notoriously noisy. An ensemble, by virtue of aggregating predictions, can provide a form of noise regularization, potentially filtering out erroneous predictions from individual models.

(3) Mitigation of Overfitting: Single models, especially deep learning architectures like LSTMs, might sometimes overfit to the training data. Ensemble methods, by averaging out the outputs of various models, can mitigate the risk of overfitting, ensuring that the classifier performs well on unseen data.

(4) Better Handling of Nonlinearities: EMG signals can display nonlinear behaviors, especially when considering dynamic muscle activities. An ensemble architecture, combining the predictions of multiple models, can better capture these nonlinear patterns, improving the classification accuracy.

(5) Increased Confidence in Predictions: With multiple models “agreeing” on a classification, there is higher confidence in the decision made by the ensemble. This can be particularly beneficial in applications where the cost of misclassification is high.

4. Data and Preprocessing

The dataset used in this study is identical to the dataset utilized in [48]. It consists of data collected from a cohort of 30 subjects. EMG signals were recorded from seven predetermined sites on the forearm and one site on the bicep. High-quality Duo-trode Ag-AgCl electrodes (Myotronics, 6140) were employed for signal acquisition. To maintain a standardized reference point, an Ag-AgCl Red-Dot electrode (3M, 2237) was placed on the wrist. The acquired signals were amplified using a Model 15 amplifier (Grass Telefactor) with a gain of 1000 and a bandwidth of 1 Hz to 1 kHz. A sampling rate of 3 kHz was used, facilitated by an analog-to-digital converter board (National Instruments, PCI-6071E). Subsequently, the EMG data were downsampled to 1 kHz before conducting the pattern classification analysis, following established protocols.

Figure 4 presents a visualization of all eight channels of one EMG signal, effectively showcasing the muscle activity during a single test trial. Each trial has consisted of the subject repeating each limb motion four times, with each motion being sustained for a duration of three seconds per repetition. The order of the limb motions was randomized to ensure unbiased data collection. To prevent any data cutoff during the recording process, a five-second rest period was introduced at the beginning and end of each trial. As a result, each trial was a duration of 94 s. A total of six trials were completed during each session, and the study encompassed four sessions conducted on separate days, ensuring a comprehensive dataset was gathered for analysis.

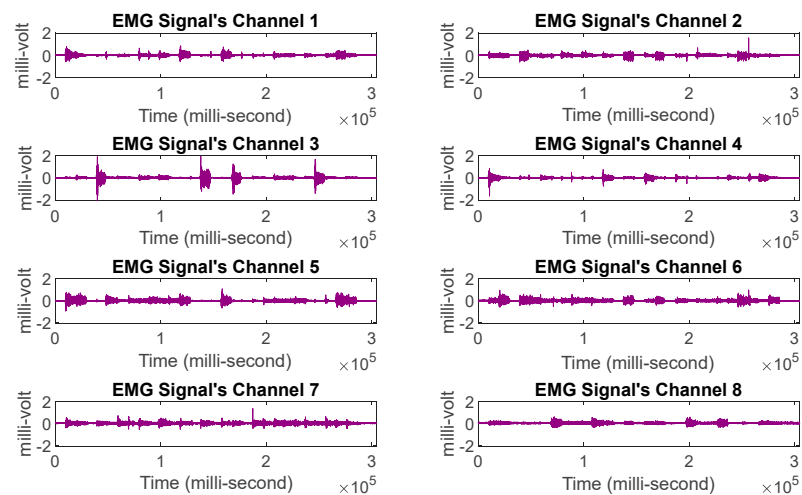


Figure 4. Eight channels of one EMG signal (sample).

In this research, a systematic process was followed to preprocess our dataset. First, we excluded extended periods of rest from both the start and end of each signal, ensuring the focus on the active portions of data. Second, we omitted the pronation and supination movements. This decision was based on the fact that the EMG did not record from the core muscles contributing to forearm pronation, and only one of the muscles involved in supination was recorded. Third, we systematically removed any remaining periods of inactivity. Fourth, we employed a bandpass filter on the signal, applying a lower cutoff frequency of 10 Hz and an upper cutoff frequency of 400 Hz. This aimed to improve the signal clarity. Fifth, we downsampled the signal and label data to a frequency of 1000 Hz to streamline the data processing. Sixth, we developed a signal mask for regions of interest (motions) and generated corresponding labels. This enabled a sequence-to-sequence classification, where each signal sample was matched with a label. Finally, in the seventh step, we divided the signals into shorter segments, each containing 12,000 samples, allowing for easier analysis.

Remark 1. *It is noteworthy that, in our methodology, we employed a bandpass filter with an upper cutoff frequency of 400 Hz, indicating our focus on signals up to this frequency. As dictated by the Nyquist theorem [49], for adequate sampling of a signal, the rate should be a minimum of twice the highest frequency present in the signal. Consequently, a signal with a frequency of 400 Hz necessitates a minimum sampling rate of 800 Hz. Subsequently, we opted to downsample to 1000 Hz, which aligns with the Nyquist theorem and is a valid approach. While this decision is not directly contradictory, there might be perceived redundancy, as filtering out frequencies up to 400 Hz and then downsampling to 1000 Hz would naturally confine us to discerning frequencies up to 500 Hz. However, this combination provides an advantage: the additional 100 Hz buffer above our bandpass filter's upper limit ensures a safeguard against potential aliasing or marginal frequency components, thereby preserving the integrity and fidelity of our signal.*

In this research, our attention was centered on four specific movements: the hand opening, hand closing, wrist flexing, and wrist extending. Our primary aim was to categorize the data in accordance with these particular actions. After the preprocessing was completed, Figure 5 shows the tagged motion areas for the first channel of the initial signal, providing a graphical depiction of the manipulated data.

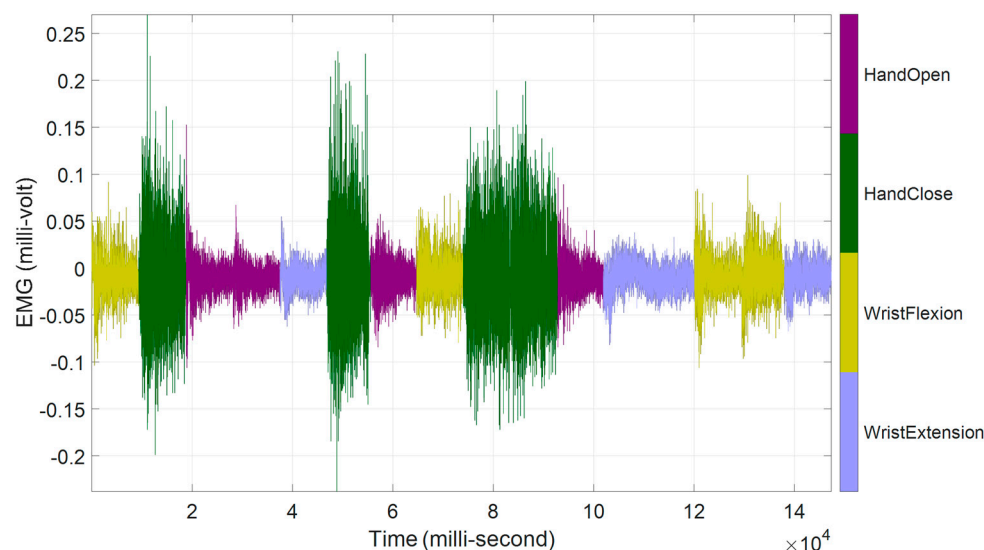


Figure 5. The processed and labeled motion regions for the first channel of the first EMG signal data.

5. Simulation Results

We divided the data into training and testing subsets, which are crucial for training the network and evaluating its performance, respectively. More specifically, we assigned 80% of the data to train the network, a step essential for adjusting the model parameters. The remaining 20% of the data was utilized for testing the network, enabling the evaluation of the model's effectiveness on data it had not encountered before.

In our proposed ensemble configuration, we employed a quartet of unique classifiers, each exhibiting a distinctive structure. The first individual classifier had an eight-node sequence input layer, a layer normalization layer, an 80-node BiLSTM layer operating in "sequence" output mode, a four-node fully connected layer, a softmax layer, and concluded with a classification layer. The second classifier mirrored the first but incorporated an LSTM projected layer with 100 hidden units, 30 cell units, and 16 output units instead of the BiLSTM layer. The third classifier echoed the previous architectures but featured a GRU (Gated Recurrent Unit) layer with 80 units in place of the LSTM projected layer. The fourth classifier replicated the first's structure but enhanced the BiLSTM layer to 120 nodes while maintaining the "sequence" output mode. For the collective ensemble network, we orchestrated a structure that commenced with a sequence input layer accommodating 24 inputs, proceeded with a layer normalization layer, an 80-node LSTM layer with attention mechanism, a four-node fully connected layer, a softmax layer, and culminated with a classification layer. This ensemble structure was tailored to leverage the advantages of each individual classifier, thus enhancing the overall model's performance.

The individual classifiers in the ensemble were strategically designed to harness the strengths of the distinct recurrent architectures: the bidirectional nature of BiLSTM for capturing forward and backward sequence information, the compactness of the projected LSTM for efficiency, and the streamlined GRU for balancing performance with fewer parameters. The variations in node configurations, like the 80- versus 120-node BiLSTM, offer differentiated modeling capacities, catering to potential sequence intricacies. In the ensemble structure, a 24-input sequence layer likely amalgamates the diverse outputs of individual classifiers. The inclusion of an 80-node LSTM with attention ensures that the ensemble concentrates on pivotal sequence segments while maintaining computational parsimony. Importantly, one of the salient benefits of our ensemble method is its resilience against architectural sensitivities; by deploying four classifiers, we are granted the flexibility to experiment with various architectures, thereby reducing the risk of over-reliance on a singular structure. This collective approach integrates the individual classifiers' advantages and boosts overall performance, while also offering a safeguard against potential architectural pitfalls.

We employed the Adam optimizer, a popular choice for training deep learning models, along with a mini-batch size of 32 to streamline the learning process. The initial learning rate was set at 0.001, providing a balanced starting point between the learning speed and model accuracy. We further capped the maximum number of epochs at 100 to avoid overfitting while ensuring sufficient training iterations. Additionally, to enhance the model's robustness and ability to generalize, we shuffled the data in each epoch.

5.1. Results of Individual Classifier and the Proposed Ensemble Technique

To illustrate the superior performance of our ensemble mechanism, we present the results of each individual classifier alongside the ensemble's output. Each of our classifiers employed state-of-the-art techniques. Thus, by evaluating the outcomes of the proposed ensemble structure, we juxtapose them with four state-of-the-art methods.

Figures 6–10 offer graphical illustrations of the training progression for each individual classifier as well as the ensemble model, charting the evolution of both the loss function and accuracy. These figures showcase both the training and validation loss and accuracy metrics. We set the batch size at 32, resulting in a total of 69 batches per epoch. The total number of epochs was set to 100. The x-axis represents the number of iterations, calculated as the product of the total number of batches and epochs. Every 20 epochs, we evaluated the improvements in accuracy for the validation data. If there was no observed improvement in the validation data accuracy over 20 consecutive epochs, we would terminate the learning process. Nonetheless, as evidenced by our figures, this condition was not met for any of the models, and the training continued up to the 100th epoch, which was the maximum number set.

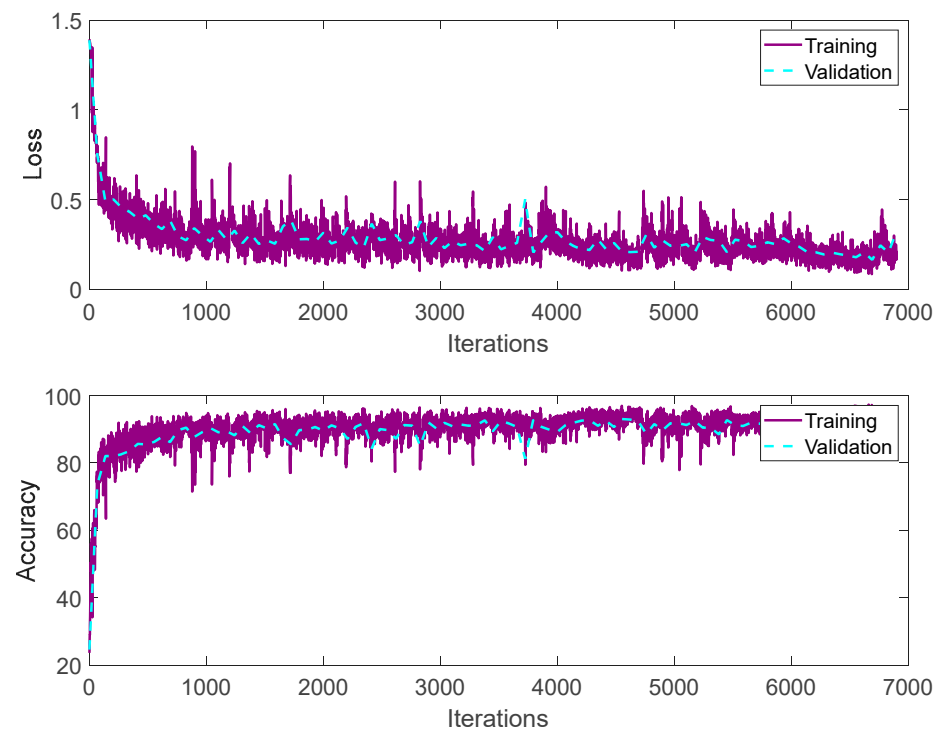


Figure 6. The loss function and accuracy of individual classifier 1 during training.

Remark 2. In machine learning, one of the primary concerns when training a model is overfitting, where the model performs exceptionally well on the training data but poorly on unseen data. One way to ascertain whether a model is overfitting is to compare its performance on the training data versus a separate validation set. In our case, the accuracy and loss function values for the validation data closely match those of the training set (see Figures 6–15). This congruence in performance is a robust indication that our model is generalizing effectively to new unseen data. Therefore, based on

this evidence, we confidently assert that our model does not suffer from overfitting. In addition, to mitigate overfitting, we instituted a check every 20 epochs to monitor accuracy improvements in the validation data. If no discernible enhancement in the validation data accuracy was noticed over these 20 consecutive epochs, the learning process was halted.

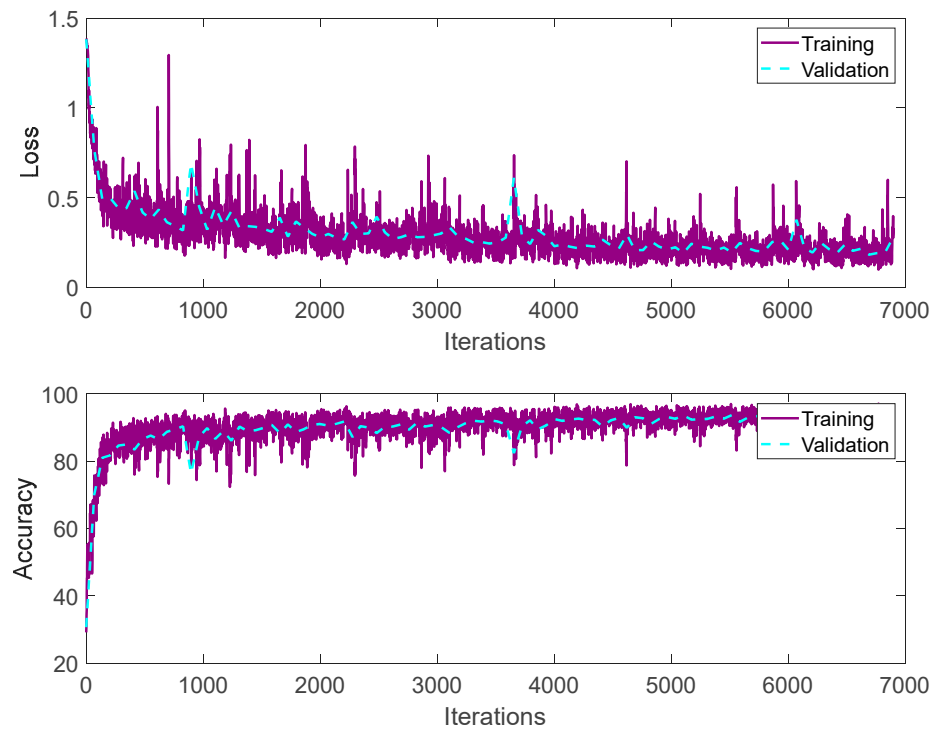


Figure 7. The loss function and accuracy of individual classifier 2 during training.

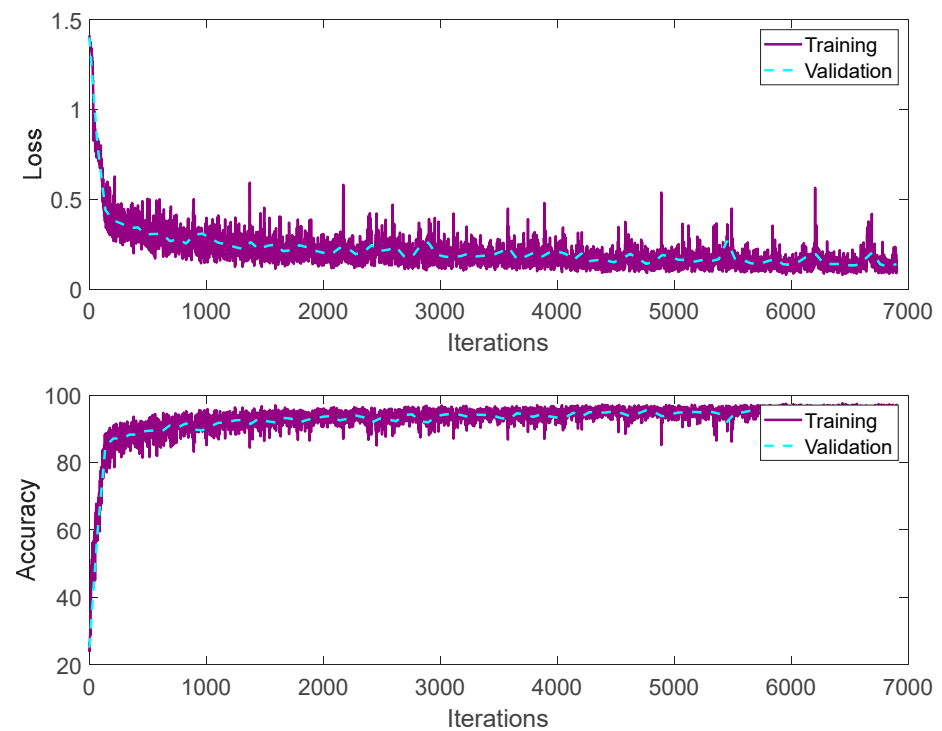


Figure 8. The loss function and accuracy of individual classifier 3 during training.

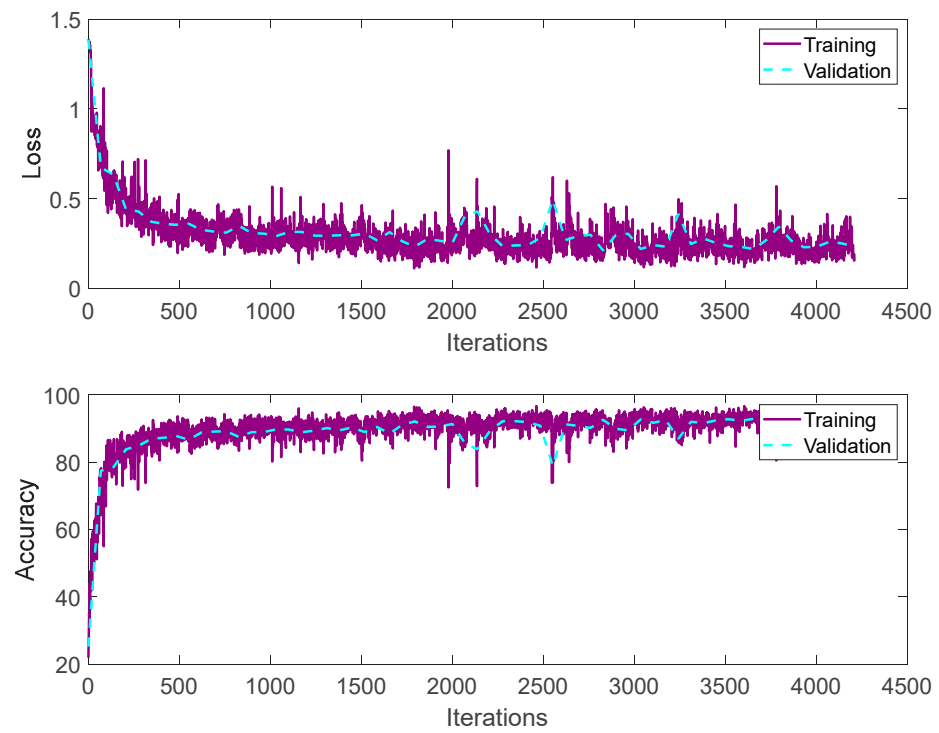


Figure 9. The loss function and accuracy of individual classifier 4 during training.

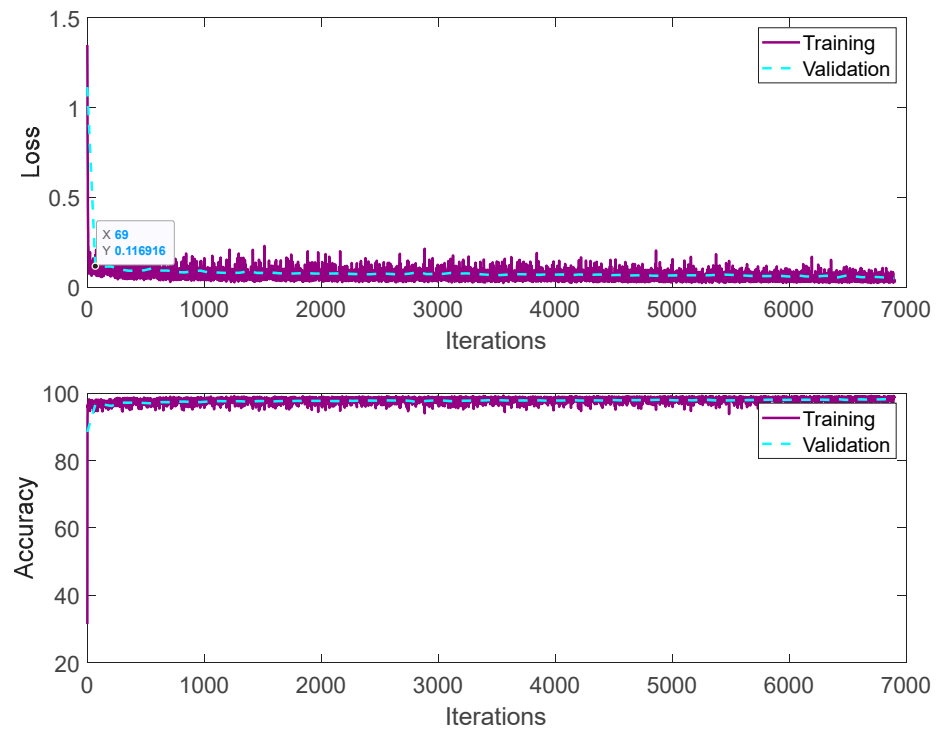


Figure 10. The loss function and accuracy of the ensemble classifier during training.

| | | | | | |
|------------|----------------|-----------------|-----------|--------------|----------------|
| True Class | HandOpen | 86.8% | 6.4% | 4.6% | 21.7% |
| | HandClose | 2.1% | 87.9% | 6.8% | 0.4% |
| | WristFlexion | 5.1% | 4.2% | 87.9% | 0.8% |
| | WristExtension | 6.0% | 1.5% | 0.7% | 77.2% |
| | | HandOpen | HandClose | WristFlexion | WristExtension |
| | | Predicted Class | | | |

Figure 11. The classification results on the testing data using individual classifier 1.

| | | | | | |
|------------|----------------|-----------------|-----------|--------------|----------------|
| True Class | HandOpen | 88.3% | 7.8% | 16.9% | 8.7% |
| | HandClose | 2.0% | 85.1% | 7.7% | 0.5% |
| | WristFlexion | 1.5% | 5.3% | 74.8% | 0.3% |
| | WristExtension | 8.2% | 1.8% | 0.7% | 90.5% |
| | | HandOpen | HandClose | WristFlexion | WristExtension |
| | | Predicted Class | | | |

Figure 12. The classification results on the testing data using individual classifier 2.

| | | | | | |
|------------|----------------|-----------------|-----------|--------------|----------------|
| True Class | HandOpen | 89.6% | 6.2% | 2.4% | 14.7% |
| | HandClose | 2.8% | 86.1% | 2.7% | 0.5% |
| | WristFlexion | 4.5% | 6.0% | 94.6% | 0.3% |
| | WristExtension | 3.1% | 1.8% | 0.3% | 84.5% |
| | | HandOpen | HandClose | WristFlexion | WristExtension |
| | | Predicted Class | | | |

Figure 13. The classification results on the testing data using individual classifier 3.

| | | | | | |
|------------|----------------|-----------------|-----------|--------------|----------------|
| True Class | HandOpen | 77.9% | 6.6% | 3.9% | 12.7% |
| | HandClose | 5.3% | 87.2% | 3.4% | 0.3% |
| | WristFlexion | 10.5% | 4.9% | 92.0% | 0.4% |
| | WristExtension | 6.4% | 1.2% | 0.6% | 86.6% |
| | | HandOpen | HandClose | WristFlexion | WristExtension |
| | | Predicted Class | | | |

Figure 14. The classification results on the testing data using individual classifier 4.

| | | | | | |
|------------|----------------|-----------------|-----------|--------------|----------------|
| True Class | HandOpen | 93.1% | 5.3% | 2.0% | 12.2% |
| | HandClose | 1.9% | 90.4% | 3.1% | 0.2% |
| | WristFlexion | 2.8% | 3.2% | 94.4% | 0.4% |
| | WristExtension | 2.2% | 1.1% | 0.6% | 87.2% |
| | | HandOpen | HandClose | WristFlexion | WristExtension |
| | | Predicted Class | | | |

Figure 15. The classification results on the testing data using the proposed technique.

Figures 11–15 offer a comprehensive visual representation, showcasing the classification outcomes for each individual classifier as well as the proposed ensemble model, all benchmarked against the testing dataset.

5.2. Using More Data for Training

To evaluate the significance of the provided data and verify the sufficiency of the training dataset, we opted not to set aside 10 percent of the data for validation. Instead, we used those samples for training, effectively utilizing 10 percent more data than was used in the simulations for Figures 11–15. Here, we only report the results of the ensemble structure. Figure 16 displays the classification results for different motions within the testing dataset using our proposed method.

| | | | | | |
|------------|----------------|-----------------|-----------|--------------|----------------|
| True Class | HandOpen | 90.3% | 5.2% | 3.0% | 7.6% |
| | HandClose | 1.7% | 91.1% | 3.7% | 0.3% |
| | WristFlexion | 5.7% | 2.4% | 92.9% | 0.5% |
| | WristExtension | 2.2% | 1.4% | 0.4% | 91.7% |
| | | HandOpen | HandClose | WristFlexion | WristExtension |
| | | Predicted Class | | | |

Figure 16. The classification results on the testing data using the proposed technique with 10% more training samples.

5.3. Comparison with an LSTM with Optimized Structure

To ensure a rigorous comparison in this paper, we juxtaposed our proposed method with an optimized LSTM. To determine the optimal structure for the LSTM, we employed Bayesian optimization. Bayesian optimization is a global optimization technique tailored for optimizing intricate, costly, and potentially noisy objective functions. It functions by creating a probabilistic model of the objective function and then leverages this model to identify the most promising evaluation candidates based on a chosen acquisition function. Gaussian Processes (GPs) are frequently adopted as models due to their adaptability and resilience in modeling unfamiliar functions. The acquisition function plays a critical role in this method, striking a balance between exploring uncharted areas of the domain and capitalizing on well-understood promising regions. For an in-depth discussion on Bayesian optimization and its framework, readers can refer to [50,51]. For finding optimal structure in neural networks like LSTM, Bayesian optimization provide substantial advantages over traditional methods like grid or random search. Bayesian optimization adopts a more structured approach, enabling the discovery of optimal or near-optimal solutions with fewer evaluations—a boon in the realm of deep learning where model training can be both computationally intensive and time-consuming.

We utilized Bayesian optimization to determine the optimal values for several hyper-parameters: the number of hidden layers in the LSTM, the initial learning rate, mini-batch size, and the number of epochs. The objective function for this optimization is defined as the negative of the accuracy achieved across all class classifications. To evaluate the performance, we set aside 10 percent of the training data. In each iteration of Bayesian optimization, the model was trained on 90 percent of the training data. Subsequently, the accuracy was assessed based on predictions for the remaining 10%. Thus, the aim of Bayesian optimization is to maximize this accuracy, which is equivalent to minimizing its negative value. We utilized the ‘expected improvement’ as the acquisition function, and the chosen kernel function was ‘Matern52’. We capped the Bayesian optimization process at 25 iterations. Figure 17 displays the progression of the objective function. As illustrated in Figure 17, the minimum objective value remained unchanged after the eighth iteration. This suggests that the Bayesian optimization identified the optimal structure within these initial eight iterations and did not find a superior structure thereafter.

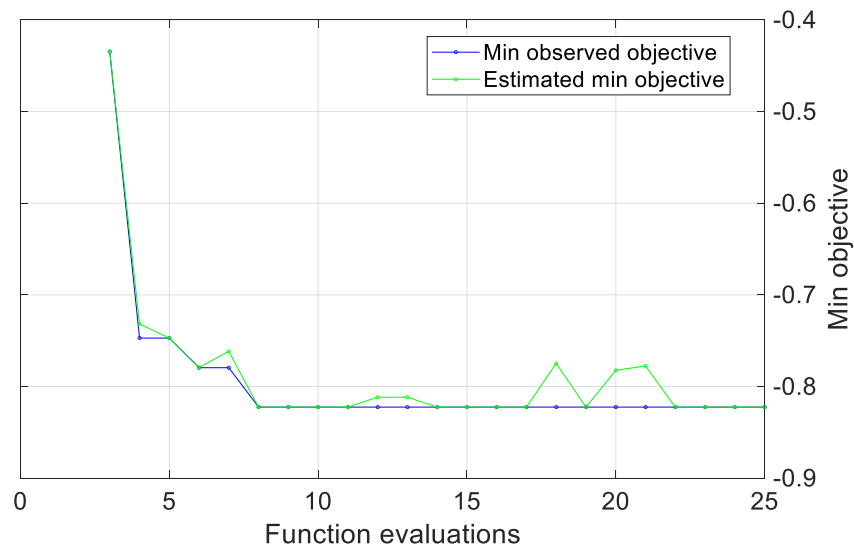


Figure 17. The objective function of Bayesian optimization for finding the optimized structure of the LSTM.

The resulting optimal structure defined by Bayesian optimization was as follows: 123 epochs, 62 hidden layers in the LSTM, a mini-batch size of 26, and an initial learning rate of 0.002713. Utilizing this optimized LSTM structure, we trained our model. The subsequent figure presents the results of the optimized LSTM on this dataset (Figure 18).

| | | | | |
|----------------|----------|-----------|--------------|----------------|
| | HandOpen | HandClose | WristFlexion | WristExtension |
| HandOpen | 85.3% | 7.1% | 7.9% | 5.5% |
| HandClose | 3.0% | 77.2% | 3.6% | 0.3% |
| WristFlexion | 1.7% | 13.1% | 88.1% | 0.2% |
| WristExtension | 10.0% | 2.6% | 0.5% | 94.0% |
| | HandOpen | HandClose | WristFlexion | WristExtension |

Figure 18. The classification results of the optimized LSTM on the test data.

The average accuracy across all classes stands at 86.15%, which is considerably lower than that achieved by our proposed ensemble, as shown in Figure 16 with the same training and testing data sets. This result illustrates that our proposed method surpasses the performance of an optimized LSTM classifier.

6. Discussion

Figures 6–10 illustrate that the accuracy and loss function values for the validation data closely align with those of the training set, providing a strong indication that overfitting did not occurred.

Figures 11–15 illustrate the nuanced performance of various classifiers; while some demonstrate proficiency in certain tasks, they fall short in others, indicating the varied strengths and weaknesses across different scenarios. As previously mentioned at the outset

of the simulation section, all classifiers utilize powerful and state-of-the-art architectures. However, they do not always yield satisfactory results. For instance:

- In Individual Classifier 1, the accuracy for all positions is below 90%. Notably, for the wrist extension position in the hand, the accuracy drops to 77.2%.
- Individual Classifier 2 performs better in detecting wrist extensions, yet its average accuracy remains at 84.67%.
- In Individual Classifier 3, only wrist flexion exceeds 90% accuracy, with all other measurements falling below this mark.
- Individual Classifier 4 detects an open hand with an accuracy of 77.9%, and its average accuracy stands at 85.92%.

Figures 11–15 reveal that while some classifiers excel in specific tasks, they underperform in others. This inconsistency underscores the advantages of our architecture: it harnesses the strengths of each classifier to yield superior results. Figure 15 substantiates this claim, with only the wrist extension position falling below 90% accuracy. All other positions, when evaluated using our proposed ensemble technique, have an accuracy exceeding 90%. The ensemble's average accuracy is 91.28%, whereas, for all individual classifiers, the average accuracy remains below 90%.

Figure 16 showcases the results of the classification process when there is an additional 10% of data available in the training process. When directly comparing the results in Figure 16 to the results in Figure 15, where we had 10% less training data, it is evident that having more training data significantly improves the detection of wrist expansion. The wrist expansion task is particularly challenging as the wrist expansion and hand open motion share the involvement of the same muscle.

In Figure 16, the average accuracy of the proposed method stands at 91.5%, with accuracies for all motions exceeding 90%. In comparison, when we allocated 10% of the data for validation, resulting in 10% less data for training, the ensemble's average accuracy was slightly reduced to 91.28%, with wrist extension accuracy being the only one below 90% at 87.2%. Hence, a close examination of both figures reveals that our approach consistently delivers commendable results under both conditions. In addition, in Section 5.3, we compared our method to an optimized LSTM. Figure 16 shows the results, with an average accuracy of 86.15% across all classes. This performance indicates our method's superiority over the optimized LSTM classifier.

7. Conclusions

We presented an innovative ensemble neural network methodology tailored for the classification of upper limb EMG signals. Our approach integrated LSTM and attention mechanisms, harnessing their respective strengths to accomplish precise classification. We provided a comprehensive description of the architecture and methodology, taking into careful consideration the distinctive attributes and obstacles associated with EMG signals. Furthermore, we outlined the preprocessing steps employed to transform the raw EMG signals into a suitable format for classification. To evaluate the effectiveness of our proposed technique, we compared its performance with a baseline LSTM classifier. The obtained numerical results demonstrated the superiority of our method in terms of accuracy and robustness, reinforcing its efficacy for EMG signal classification. The average accuracy of our proposed method surpasses the 90% threshold. In contrast, the individual classifiers tend to lag behind, with their average accuracy falling below 90%. For some specific motion detections, their performance even approaches a lower boundary near 75%. As a future suggestion, the proposed technique could be integrated and investigated in a probabilistic manner to further enhance accuracy and reliability. This direction of research holds promise for advancing EMG signal classification.

Author Contributions: Conceptualization, N.D.A., H.J., Q.Y., J.M. and S.B.; Methodology, N.D.A., H.J., Q.Y., J.M. and S.B.; Validation, N.D.A., H.J., Q.Y., J.M. and S.B.; Formal analysis, N.D.A., H.J., Q.Y., J.M. and S.B.; Writing—original draft, N.D.A., H.J., Q.Y., J.M. and S.B.; Writing—review &

editing, N.D.A., H.J., Q.Y., J.M. and S.B.; Supervision, N.D.A., H.J., Q.Y., J.M. and S.B. All authors have read and agreed to the published version of the manuscript.

Funding: This research work was funded by Institutional Fund Projects under grant no. (IFPIP: 78-135-1443). The authors gratefully acknowledge technical and financial support provided by the Ministry of Education and King Abdulaziz University, DSR, Jeddah, Saudi Arabia.

Data Availability Statement: Not Applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

- De Luca, C. *Electromyography. Encyclopedia of Medical Devices and Instrumentation*; John Wiley Publisher: Hoboken, NJ, USA, 2006.
- Gohel, V.; Mehendale, N. Review on Electromyography Signal Acquisition and Processing. *Biophys. Rev.* **2020**, *12*, 1361–1367. [[CrossRef](#)] [[PubMed](#)]
- Norali, A.N.; Som, M.; Kangar-Arau, J. Surface Electromyography Signal Processing and Application: A Review. In Proceedings of the International Conference on Man-Machine Systems (ICoMMS), Batu Ferringhi, Malaysia, 11–13 October 2009.
- Jiang, Y.; Chen, C.; Zhang, X.; Chen, C.; Zhou, Y.; Ni, G.; Muh, S.; Lemos, S. Shoulder Muscle Activation Pattern Recognition Based on SEMG and Machine Learning Algorithms. *Comput. Methods Programs Biomed.* **2020**, *197*, 105721. [[CrossRef](#)] [[PubMed](#)]
- Ahsan, M.R.; Ibrahimy, M.I.; Khalifa, O.O. EMG Signal Classification for Human Computer Interaction: A Review. *Eur. J. Sci. Res.* **2009**, *33*, 480–501.
- Zecca, M.; Micera, S.; Carrozza, M.C.; Dario, P. Control of Multifunctional Prosthetic Hands by Processing the Electromyographic Signal. *Crit. Rev. Biomed. Eng.* **2002**, *30*, 459–485. [[CrossRef](#)] [[PubMed](#)]
- Rittenhouse, D.M.; Abdullah, H.A.; Runciman, R.J.; Basir, O. A Neural Network Model for Reconstructing EMG Signals from Eight Shoulder Muscles: Consequences for Rehabilitation Robotics and Biofeedback. *J. Biomech.* **2006**, *39*, 1924–1932. [[CrossRef](#)]
- Shabani, A.; Mahjoob, M.J. Bio-Signal Interface for Knee Rehabilitation Robot Utilizing EMG Signals of Thigh Muscles. In Proceedings of the 2016 4th International Conference on Robotics and Mechatronics (ICROM), Tehran, Iran, 26–28 October 2016; IEEE: Piscataway, NJ, USA, 2016; pp. 228–233.
- Han, J.-S.; Song, W.-K.; Kim, J.-S.; Bang, W.-C.; Lee, H.; Bien, Z. New EMG Pattern Recognition Based on Soft Computing Techniques and Its Application to Control of a Rehabilitation Robotic Arm. In Proceedings of the 6th International Conference on Soft Computing (IIZUKA2000), Fukuoka, Japan, 1–4 October 2000; pp. 890–897.
- Jafarzadeh, M.; Hussey, D.C.; Tadesse, Y. Deep Learning Approach to Control of Prosthetic Hands with Electromyography Signals. In Proceedings of the 2019 IEEE International Symposium on Measurement and Control in Robotics (ISMCR), Houston, TX, USA, 19–21 September 2019; IEEE: Piscataway, NJ, USA, 2019; pp. A1–A4.
- Unanyan, N.N.; Belov, A.A. Design of Upper Limb Prosthesis Using Real-Time Motion Detection Method Based on EMG Signal Processing. *Biomed. Signal Process. Control* **2021**, *70*, 103062. [[CrossRef](#)]
- Felici, F.; Del Vecchio, A. Surface Electromyography: What Limits Its Use in Exercise and Sport Physiology? *Front. Neurol.* **2020**, *11*, 578504. [[CrossRef](#)]
- Zhou, T.; Omisore, O.M.; Du, W.; Duan, W.; Zhang, Y.; Wang, L. *A Preliminary Study on Surface Electromyography Signal Analysis for Motion Characterization during Catheterization*; Springer: Berlin/Heidelberg, Germany, 2019; pp. 617–628.
- Tepe, C.; Demir, M. Real-Time Classification of EMG Myo Armband Data Using Support Vector Machine. *IRBM* **2022**, *43*, 300–308. [[CrossRef](#)]
- Abu, M.A.; Rosleesham, S.; Suboh, M.Z.; Yid, M.S.M.; Kornain, Z.; Jamaluddin, N.F. Classification of EMG Signal for Multiple Hand Gestures Based on Neural Network. *Indones. J. Electr. Eng. Comput. Sci.* **2020**, *17*, 256–263.
- Khan, M.U.; Aziz, S.; Bilal, M.; Aamir, M.B. Classification of EMG Signals for Assessment of Neuromuscular Disorder Using Empirical Mode Decomposition and Logistic Regression. In Proceedings of the 2019 International Conference on Applied and Engineering Mathematics (ICAEM), Taxila, Pakistan, 27–29 August 2019; IEEE: Piscataway, NJ, USA, 2019; pp. 237–243.
- Yassin, M.M.; Saber, A.M.; Saad, M.N.; Said, A.M.; Khalifa, A.M. Developing a Low-Cost, Smart, Handheld Electromyography Biofeedback System for Telerehabilitation with Clinical Evaluation. *Med. Nov. Technol. Devices* **2021**, *10*, 100056. [[CrossRef](#)]
- Ay, A.N.; Yildiz, M.Z. The Performance of an Electromyography-based Deep Neural Network Classifier for External and Internal Focus Instructions. *Concurr. Comput. Pract. Exp.* **2023**, *35*, e7470. [[CrossRef](#)]
- Fang, C.; He, B.; Wang, Y.; Cao, J.; Gao, S. EMG-Centered Multisensory Based Technologies for Pattern Recognition in Rehabilitation: State of the Art and Challenges. *Biosensors* **2020**, *10*, 85. [[CrossRef](#)] [[PubMed](#)]
- Dempster, A. A Very Fast (Almost) Deterministic Transform for Time Series Classification Angus Dempster, Daniel F. Schmidt, Geoffrey I. Webb. *arXiv* **2020**, arXiv:2012.08791.
- Iwana, B.K.; Uchida, S. An Empirical Survey of Data Augmentation for Time Series Classification with Neural Networks. *PLoS ONE* **2021**, *16*, e0254841. [[CrossRef](#)]
- Hatami, N.; Gavet, Y.; Debayle, J. *Classification of Time-Series Images Using Deep Convolutional Neural Networks*; SPIE: Bellingham, WA, USA, 2018; Volume 10696, pp. 242–249.

23. Graves, A.; Graves, A. Long Short-Term Memory. In *Supervised Sequence Labelling with Recurrent Neural Networks*; Springer: Berlin/Heidelberg, Germany, 2012; pp. 37–45.
24. Schmidhuber, J.; Hochreiter, S. Long Short-Term Memory. *Neural Comput.* **1997**, *9*, 1735–1780.
25. Livieris, I.E.; Pintelas, E.; Pintelas, P. A CNN–LSTM Model for Gold Price Time-Series Forecasting. *Neural Comput. Appl.* **2020**, *32*, 17351–17360. [[CrossRef](#)]
26. Tang, Y.; Yu, F.; Pedrycz, W.; Yang, X.; Wang, J.; Liu, S. Building Trend Fuzzy Granulation-Based LSTM Recurrent Neural Network for Long-Term Time-Series Forecasting. *IEEE Trans. Fuzzy Syst.* **2021**, *30*, 1599–1613. [[CrossRef](#)]
27. Tigrini, A.; Scattolini, M.; Mengarelli, A.; Fioretti, S.; Morettini, M.; Burattini, L.; Verdini, F. Role of the Window Length for Myoelectric Pattern Recognition in Detecting User Intent of Motion. In Proceedings of the 2022 IEEE International Symposium on Medical Measurements and Applications (MeMeA), Messina, Italy, 22–24 June 2022; IEEE: Piscataway, NJ, USA, 2022; pp. 1–6.
28. Ding, Z.; Yang, C.; Tian, Z.; Yi, C.; Fu, Y.; Jiang, F. SEMG-Based Gesture Recognition with Convolution Neural Networks. *Sustainability* **2018**, *10*, 1865. [[CrossRef](#)]
29. Nasri, N.; Orts-Escolano, S.; Gomez-Donoso, F.; Cazorla, M. Inferring Static Hand Poses from a Low-Cost Non-Intrusive SEMG Sensor. *Sensors* **2019**, *19*, 371. [[CrossRef](#)]
30. Xiong, D.; Zhang, D.; Zhao, X.; Zhao, Y. Deep Learning for EMG-Based Human-Machine Interaction: A Review. *IEEE/CAA J. Autom. Sin.* **2021**, *8*, 512–533. [[CrossRef](#)]
31. Li, W.; Shao, K.; Zhu, C.; Ma, Y.; Cao, W.; Yin, M.; Yang, L.; Luo, M.; Wu, X. Preliminary Study of Online Real-Time Control System for Lower Extremity Exoskeletons Based on EEG and SEMG Fusion. In Proceedings of the 2022 IEEE International Conference on Robotics and Biomimetics (ROBIO), Jinghong, China, 5–9 December 2022; IEEE: Piscataway, NJ, USA, 2022; pp. 1689–1694.
32. Zhou, X.; Wang, C.; Zhang, L.; Liu, J.; Liang, G.; Wu, X. Continuous Estimation of Lower Limb Joint Angles From Multi-Stream Signals Based on Knowledge Tracing. *IEEE Robot. Autom. Lett.* **2023**, *8*, 951–957. [[CrossRef](#)]
33. Tigrini, A.; Al-Timemy, A.H.; Verdini, F.; Fioretti, S.; Morettini, M.; Burattini, L.; Mengarelli, A. Decoding Transient SEMG Data for Intent Motion Recognition in Transhumeral Amputees. *Biomed. Signal Process. Control* **2023**, *85*, 104936. [[CrossRef](#)]
34. Srivastava, A. Classification of Emg Signals of Eye Movement Using Machine Learning Techniques. Available online: <http://dspace.dtu.ac.in:8080/jspui/handle/repository/19867> (accessed on 31 May 2023).
35. Pilkar, R.; Momeni, K.; Ramanujam, A.; Ravi, M.; Garbarini, E.; Forrest, G.F. Use of Surface Emg in Clinical Rehabilitation of Individuals with Sci: Barriers and Future Considerations. *Front. Neurol.* **2020**, *11*, 578559. [[CrossRef](#)] [[PubMed](#)]
36. Cheng, M.; Sori, W.J.; Jiang, F.; Khan, A.; Liu, S. Recurrent Neural Network Based Classification of ECG Signal Features for Obstruction of Sleep Apnea Detection. In Proceedings of the 2017 IEEE International Conference on Computational Science and Engineering (CSE) and IEEE International Conference on Embedded and Ubiquitous Computing (EUC), Guangzhou, China, 21–24 July 2017; IEEE: Piscataway, NJ, USA, 2017; Volume 2, pp. 199–202.
37. Naul, B.; Bloom, J.S.; Pérez, F.; van der Walt, S. A Recurrent Neural Network for Classification of Unevenly Sampled Variable Stars. *Nat. Astron.* **2018**, *2*, 151–155. [[CrossRef](#)]
38. Simão, M.; Neto, P.; Gibaru, O. EMG-Based Online Classification of Gestures with Recurrent Neural Networks. *Pattern Recognit. Lett.* **2019**, *128*, 45–51. [[CrossRef](#)]
39. Ma, Y.; Donati, E.; Chen, B.; Ren, P.; Zheng, N.; Indiveri, G. Neuromorphic Implementation of a Recurrent Neural Network for EMG Classification. In Proceedings of the 2020 2nd IEEE International Conference on Artificial Intelligence Circuits and Systems (AICAS), Genova, Italy, 31 August–2 September 2020; IEEE: Piscataway, NJ, USA, 2020; pp. 69–73.
40. Welch, R.L.; Ruffing, S.M.; Venayagamoorthy, G.K. Comparison of Feedforward and Feedback Neural Network Architectures for Short Term Wind Speed Prediction. In Proceedings of the 2009 International Joint Conference on Neural Networks, Atlanta, GA, USA, 14–19 June 2009; IEEE: Piscataway, NJ, USA, 2009; pp. 3335–3340.
41. Sundermeyer, M.; Oparin, I.; Gauvain, J.-L.; Freiberg, B.; Schlüter, R.; Ney, H. Comparison of Feedforward and Recurrent Neural Network Language Models. In Proceedings of the 2013 IEEE International Conference on Acoustics, Speech and Signal Processing, Vancouver, BC, Canada, 26–31 May 2013; IEEE: Piscataway, NJ, USA, 2013; pp. 8430–8434.
42. Men, L.; Ilk, N.; Tang, X.; Liu, Y. Multi-Disease Prediction Using LSTM Recurrent Neural Networks. *Expert Syst. Appl.* **2021**, *177*, 114905. [[CrossRef](#)]
43. Weerakody, P.B.; Wong, K.W.; Wang, G.; Ela, W. A Review of Irregular Time Series Data Handling with Gated Recurrent Neural Networks. *Neurocomputing* **2021**, *441*, 161–178. [[CrossRef](#)]
44. Farzad, A.; Mashayekhi, H.; Hassanpour, H. A Comparative Performance Analysis of Different Activation Functions in LSTM Networks for Classification. *Neural Comput. Appl.* **2019**, *31*, 2507–2521. [[CrossRef](#)]
45. Wang, Y.; Huang, M.; Zhu, X.; Zhao, L. Attention-Based LSTM for Aspect-Level Sentiment Classification. In Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, Austin, TX, USA, 1–5 November 2016; pp. 606–615.
46. Liu, G.; Guo, J. Bidirectional LSTM with Attention Mechanism and Convolutional Layer for Text Classification. *Neurocomputing* **2019**, *337*, 325–338. [[CrossRef](#)]
47. Lee, X.Y.; Kumar, A.; Vidyaratne, L.; Rao, A.R.; Farahat, A.; Gupta, C. An Ensemble of Convolution-Based Methods for Fault Detection Using Vibration Signals. *arXiv* **2023**, arXiv:2305.05532.
48. Goge, A.R.; Chan, A. Investigating Classification Parameters for Continuous Myoelectrically Controlled Prostheses. *CMBES Proc.* **2005**, *28*, 1–4.

49. Kipnis, A.; Goldsmith, A.J.; Eldar, Y.C.; Weissman, T. Distortion Rate Function of Sub-Nyquist Sampled Gaussian Sources. *IEEE Trans. Inf. Theory* **2015**, *62*, 401–429. [[CrossRef](#)]
50. Garnett, R. *Bayesian Optimization*; Cambridge University Press: Cambridge, UK, 2023; ISBN 1-108-62355-7.
51. Frazier, P.I. A Tutorial on Bayesian Optimization. *arXiv* **2018**, arXiv:1807.02811.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.