



ARTICLE

A Novel Locomotion Rule Rmbedding Long Short-Term Memory Network with Attention for Human Locomotor Intent Classification Using Multi-Sensors Signals

Jiajie Shen¹, Yan Wang^{1,*} and Dongxu Zhang²

¹Key Laboratory of Symbol Computation and Knowledge Engineering, Ministry of Education, Colleague of Computer Science and Technology, Jilin University, Changchun, 130012, China

²College of Software, and Key Laboratory of Symbolic Computation and Knowledge Engineering of Ministry of Education, Jilin University, Changchun, 130012, China

*Corresponding Author: Yan Wang. Email: wy6868@jlu.edu.cn

Received: 21 November 2023 Accepted: 11 April 2024 Published: 20 June 2024

ABSTRACT

Locomotor intent classification has become a research hotspot due to its importance to the development of assistive robotics and wearable devices. Previous work have achieved impressive performance in classifying steady locomotion states. However, it remains challenging for these methods to attain high accuracy when facing transitions between steady locomotion states. Due to the similarities between the information of the transitions and their adjacent steady states. Furthermore, most of these methods rely solely on data and overlook the objective laws between physical activities, resulting in lower accuracy, particularly when encountering complex locomotion modes such as transitions. To address the existing deficiencies, we propose the locomotion rule embedding long short-term memory (LSTM) network with Attention (LREAL) for human locomotor intent classification, with a particular focus on transitions, using data from fewer sensors (two inertial measurement units and four goniometers). The LREAL network consists of two levels: One responsible for distinguishing between steady states and transitions, and the other for the accurate identification of locomotor intent. Each classifier in these levels is composed of multiple-LSTM layers and an attention mechanism. To introduce real-world motion rules and apply constraints to the network, a prior knowledge was added to the network via a rule-modulating block. The method was tested on the ENABL3S dataset, which contains continuous locomotion data for seven steady and twelve transitions states. Experimental results showed that the LREAL network could recognize locomotor intents with an average accuracy of 99.03% and 96.52% for the steady and transitions states, respectively. It is worth noting that the LREAL network accuracy for transition-state recognition improved by 0.18% compared to other state-of-the-art network, while using data from fewer sensors.

KEYWORDS

Lower-limb prosthetics; deep neural networks; motion classification



1 Introduction

Millions of people globally suffer from severe disabilities, which can drastically decline the quality of life [1]. Fortunately, with the development of robotics and wearable sensor technology, disabled people can now live relatively comfortably with the help of wearable assistive devices [2,3]. To work effectively and ensure the safety of users, devices such as wearable robots or exoskeletons need to recognize the locomotor intent of the user precisely. However, it can be difficult for the devices to distinguish human locomotor intent as it can not be observed directly. Therefore wearable robots identify locomotion by using real-time data from wearable sensors, such as electromyography (EMG) [3], inertial measurement units (IMUs) [4], load cell [5], pressure sensor [6], goniometer (GONIO) [7], and fused signals from multiple sources [8].

Several studies have attempted to enable human locomotor intent detection by robots, via traditional machine learning (ML) approaches, such as support vector machines (SVMs) [9], linear discriminant analysis (LDA) [10], quadratic discriminant analysis (QDA) [11], and artificial neural networks (ANNs) [12] and achieved high prediction accuracies. While these methods can recognize locomotor intent successfully, they require extensive feature engineering in advance. Moreover, when the situation becomes complex or the decision boundary is fuzzy when facing transitions, the traditional ML method performances decline [13]. Compared to classical ML methods, deep learning (DL) methods, such as convolutional neural networks (CNNs) [13], recurrent neural networks (RNNs) [14], gated recurrent units (GRUs) [15], can predict complex locomotion modes with minimal feature engineering. Recently, deep reinforcement learning (DRL) has garnered significant attention as an interactive ML paradigm that seamlessly integrates deep neural networks (DNNs) [16] into the well-established conventional reinforcement learning (RL) framework [17]. Many DRL approaches, such as the Markov decision process (MDP) and self-organizing networks have obtained remarkable achievements in several fields [18–20], offering new possibilities and advancements in the field of lower-limb prosthetics. However, RL typically requires a large number of interaction samples for training, which can be a challenge in the real world, it still needs further development before being applied to practical scenarios.

Using the time history information on locomotion sequence, long short-term memory (LSTM) networks have demonstrated good results for classifying human locomotion [21]. LSTM concentrates on the long-range information of the previous moments for accurate recognition, especially on steady states. However, this method may not perform as well when facing transitions as they are often situated between two steady states, and most of their historical information would be similar to that of the adjacent state. As a result, raw LSTM may not be able to focus on crucial time instances to accurately classify transition modes.

Attention mechanism (AM) has been an emerging research direction that has been applied to several fields such as natural language processing (NLP) and computer vision in recent years for its ability to focus on key parts of temporal information [22]. However, AM has not yet been widely used or explored in human locomotion mode prediction. Zhu et al. [23] proposed a knee/ankle joint angle prediction method based on attention-based CNN-LSTM model. They deployed the AM before the LSTM layers to extract detailed information from the CNN while suppressing useless information. As the attention layer in this model is located before LSTM layers, the structure of the latter becomes N-to-1, i.e., the LSTM layers obtain a result directly from the input. Therefore, the AM is mainly used for spatial feature selection rather than temporal characteristics.

Despite their advantages, most DNNs for human locomotor prediction are purely data-driven, and lack an understanding of locomotion rules. This can lead to low model accuracy, especially when

meeting complex locomotion modes (e.g., transitions). Rule-based methods such as state machines [24] and decision trees have been applied to address the issue. However, to achieve high accuracies similar to that of DNNs, rule-based methods require a deep understanding of the experimental data to obtain the rules. Rule designing becomes increasingly complex when meeting highly diverse locomotion modes. Considering the respective advantages of DNNs and rule-based methods, it may be helpful to combine a priori knowledge with the former for the optimal utilization of the two.

In response to the existing deficiencies, in this study, we proposed a locomotion rule embedding LSTM network with AM (LREAL), a system to identify human locomotor intent precisely, based on IMU and GONIO sensor signals. We introduced the AM after the LSTM layer to redistribute the attention of different features at different times. This creates an N-to-N LSTM layer struct, which is maintained by its the output. This attention layer placement allows the model to focus on the complete spatiotemporal characteristics. To drive the network by using both data and rules (instead of only data), inspired by rule-embedded neural networks (ReNN) [25], we introduced the locomotion information to the model by embedding the rule in the network. Specifically, the LREAL network is composed of two levels: The first is a steady/transition state classifier and the second is composed of two classifiers to accurately recognize the locomotor intent of the states. Each classifier in these levels is combined with LSTM layers and an attention layer. Initially, continuous raw IMU and GONIO data are processed into fixed-size fragments for input into the structure. After the first level of classification, the fragments from steady locomotion modes are input into the classifier for steady locomotion to classify steady locomotor intent. It further passes a result in the form of a probability vector to a rule modulating block. In this block, the rule will be combined with the probability vector to obtain an encoding vector for transitions. As transitions follow steady locomotion, the features obtained by the rule-modulating block are directly input into the attention layer of the transitions-state classifier for prediction. The proposed model performance vastly improved human locomotor intent recognition compared to CNN, LSTM, and other ML methods. The contributions of this study can be summarized as follows:

- We introduced AM to the LSTM network, enhancing the ability of the model to capture key instances in time and effectively extract locomotion-related features of the sensor data.
- We combined LSTM layers and an AM to build a classifier (AT-LSTM) that enables complete learning from two IMU and four GONIO datasets.
- We introduced a priori knowledge via a rule modulating block to the network to constrain the decision scope when meeting the transition states, and demonstrate the added effectiveness via ablation studies.
- We constructed the LREAL work with a multi-level architecture composed of AT-LSTM for human locomotor intent and validated its efficacy using ENABL3S dataset [26] through accuracy scores. Our results using fewer sensors were comparable to or better than the existing models.

2 Literature Review

In this section, we discuss the related works and technologies on: Sensors, locomotor intent classification and ReNNs.

2.1 Sensors

Sensors play a crucial role in the design of exoskeletons and wearable robots as they greatly influence reliability and cost. Wearable robots most commonly use IMU sensors which record 3-axis

acceleration, angular velocity, and gyroscope signals [4,15]. IMUs are preferred as they provide reliable data and are cost-effective. EMG sensors are employed for human activity recognition (HAR) as they can measure electrical signals generated from muscle contractions during physical activities [3]. While EMG data can significantly enhance their performance wearable robots are relatively expensive and inconvenient to wear. Other types of sensors, such as load cells, pressure sensors, (GONIO), are also used. Furthermore, the fusion of signals from multiple sources has also been widely applied to leverage their respective advantages [8]. The selection of sensor types and data processing methods can greatly impact the locomotor intent classifier performance, and finding ways to achieve better results with fewer sensors has been the primary research focal point.

2.2 Locomotor Intent Classification

Motivated by their successful application in various fields, DL methods have gained significant popularity in locomotor intent classification. CNNs, known for their effective feature learning in CV, have been applied to locomotor intent classification by similarly processing sensor data [27]. While CNNs offer advantages such as parameter sharing and sparse connections, they require higher computational resources. LSTM networks, known for their ability to process time series data, have shown promising results in classifying human locomotion [21]. To enhance efficiency, studies have explored CNN-RNN hybrid networks. Wang et al. [28] proposed a CNN-LSTM hybrid approach, achieving a validation accuracy of 95.90% in recognizing six types of activities (walking, lying, sitting, standing, and stair ascent, and stair descent) using a smartphone IMU. Zhu et al. [23] proposed an attention-based CNN-LSTM model for knee/ankle joint angle movement prediction, using AM for feature extraction. In contrast to [23], the AM in LREAL was introduced after the LSTM layer to dynamically redistribute attention across different features at different time steps, aiming for complete spatiotemporal feature selection.

2.3 ReNN

The interpretability of DNNs is often criticized due to the difficulty in explaining inferences as concise interactions among parameters and the network [29]. Unlike human inferences based on experiences and knowledge, most DNNs are purely data-driven. This lack of prior knowledge can result in low interpretability and potential errors in the models [30]. To address these issues, knowledge representation has been utilized to describe the richness of the world in computer systems, allowing artificial intelligence to understand and utilize it for reasoning and inference. Rule-based representation, a common formalism of knowledge representation, has been widely employed in expert systems and achieved excellent results [31]. Considering the aforementioned challenges and inspired by the concept of rules embedded in neural networks [25], we propose a rule-modulating block that incorporates locomotion rules into a knowledge representation and combines it with DNNs. This approach aims to enhance interpretability and improve the model performance by leveraging both data-driven and rule-based learning.

3 Dataset and Preprocessing Steps

3.1 Dataset

We used the ENABL3S public benchmark dataset to validate the proposed method. The dataset consists of data from IMU sensors (placed on the thigh and shank of the subjects), GONIO sensors (placed on knee and ankle of the subjects) and EMG sensors, collected from 10 able-bodied subjects (seven males and three females) with an average age of (25.5 ± 2) y, height of (174 ± 12) cm, and weight

of (70 ± 14) kg. The subjects move according to the required mode [sit (S), stand (ST), ground-level walking (LW), ramp ascending/descending (RA/RD), and stair ascent/descent (SA/SD)]. The ramps have a 10° slope and the stairs consist of four steps. The odd-numbered moving sequence is: $S \rightarrow ST \rightarrow LW \rightarrow SA \rightarrow LW \rightarrow RD \rightarrow LW \rightarrow ST \rightarrow S$; and the even-numbered moving sequence is: $S \rightarrow ST \rightarrow LW \rightarrow RA \rightarrow LW \rightarrow SD \rightarrow LW \rightarrow ST \rightarrow S$. There are seven kinds of steady locomotion modes in ENABL3S with recorded true labels. Each data sample contains the signals from five IMU sensors recording six data channels (3 axis acceleration and angular velocity) and four GONIO sensors. By using a key fob, the start and end of the steady locomotion modes are recorded and the data is labeled according to the true moving mode. Although only labels of steady locomotion modes are available in ENABL3S, the information on the transition modes can be obtained by extracting the data between the steady modes as ENABL3S marks the time of the beginning and ends of the steady states.

3.2 Data Preprocessing

The IMU data underwent low-pass filtering at 20 Hz to eliminate high-frequency noise. In the ENABL3S dataset, the IMU and GONIO data are sampled at 500 Hz (i.e., sampling once every 2 ms). Every 250 samples (2 ms per sample) were segmented into a 500 ms analysis window with a sliding window of 50 ms. We used the locomotion mode at the end of each gait event to obtain the label of each analysis window. If the beginning and end of the gait event maintained the same locomotion mode, analysis windows for the gait event were directly labeled as “steady locomotion intent” (Fig. 1). Gait events with different modes can experience two different situations. If the analysis window was located (i) fully within a transition gait event, it was labeled according to the starting and ending mode and (ii) partially within a transition gait event of transitions, it was labeled based on the majority of its location ($>50\%$; Table 1). We utilized the acceleration and angular velocity data of only two IMU sensors on the upper and lower right leg in 3 axes (X, Y and Z axes) and GONIO data, so the initial input of the network can be denoted as $x \in R^{500 \times 12}$. The above data was divided as follow: 90% for training and 10% for testing. Additionally, 20% of the training set was also used for validation to prevent the network from over-fitting.

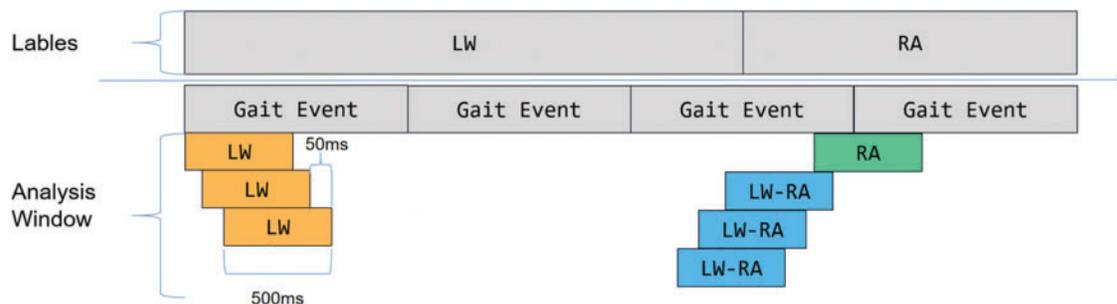


Figure 1: Sequence extraction visualization and labeling: The yellow windows refer to the steady LW locomotor intent, as they are located fully within a steady gait event; the blue windows refer to the LW-RA transition, as they are located fully within a transition gait event; and the green window refers to steady RA locomotor intent as the start and end of the window fails in the same locomotor mode

Table 1: Analysis windows extracted for each label from the ENABL3S dataset; each window is composed of 250 samples and each sample has 12 features

Label	Number	Label	Number
S	29089	SA-LW	1158
LW	45741	LW-RD	1805
RA	15346	RD-LW	1420
RD	17730	LW-ST	1556
SA	5641	ST-S	2065
SD	5499	LW-RA	1778
ST	6552	RA-LW	1410
S-ST	8282	LW-SD	1638
ST-LW	4045	SD-LW	1392
LW-SA	1390		

4 Methods

4.1 Network Architecture

The proposed LREAL network is composed of two levels (Fig. 2). The first level, i.e., the steady/transition classifier, is responsible for initially classifying the input sequence and dividing it into the two main states. This classifier treats both steady and transition states equally. Its purpose is to serve as a pre-classification step for the subsequent level.

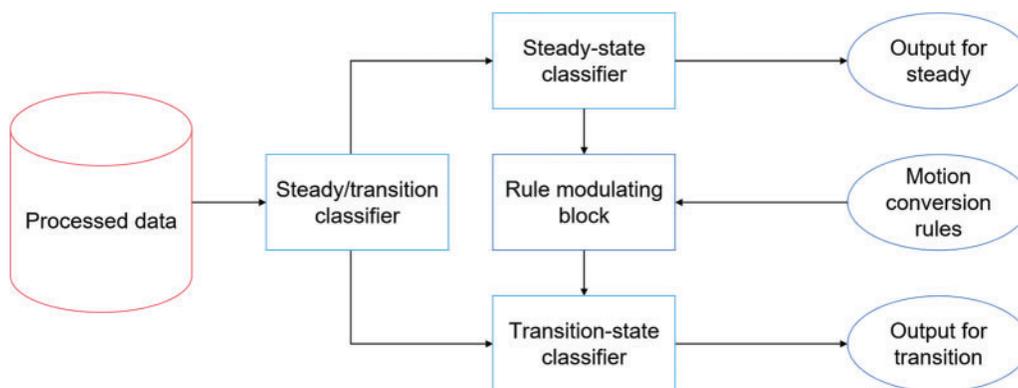


Figure 2: Schematic representation LREAL network for human locomotor intent recognition

The second level consists of two distinct classifiers and a rule-modulating block. The input sequence divided by the first-level classifier is directed to the appropriate classifier based on the assigned state label. The steady-state classifier receives the fragments from the steady-state portion of the sequence as its input. The output of this classifier is stored and passed onto the rule-modulating block as auxiliary information, which helps in preparing for the recognition of transitions.

As mentioned previously, transitions occur between steady states, i.e., there always exists a preceding steady state that significantly influences the range of post-transition behavior. There also exist specific rules governing the motion transformations, such as considering the previous locomotion

mode when encountering a transition [32]. Incorporating these rules and prior information from the classifier for steady locomotion into the transition-state classifier does not affect practical applications or add computational cost. Therefore, the rule-modulating block is designed to introduce motion transformation rules to the network. It receives transmission information from the steady-state classifier to obtain the preceding condition for each window for transitions. This information is combined with the motion transformation rules to assist the transition-state classifier in recognition. This process embeds the provided motion conversion rules with auxiliary information from the preceding steady state, resulting in an output vector that represents refined information.

Finally, the transition-state classifier uses the sequence of recognized transitions from the first-level classifier and refined information from the rule-modulating block to generate the final output. Note that the classification of steady states solely relies on the data, while the transition recognition combines the data with rule-embedded information. This integration of data and rule-based information enhances the of transition recognition accuracy.

4.2 AM Design

As different samples within a specific analysis window may contribute differently to the recognition of various activities, the proposed AT-LSTM model employs an AM to capture the relationship between the samples and locomotion classes (Fig. 3). The attention layer first derives a vector representation for the features of each sample and maps to a probability value indicating the likelihood of the analysis window belonging to a certain locomotion class. Formally, we constructed the attention layer as follows.

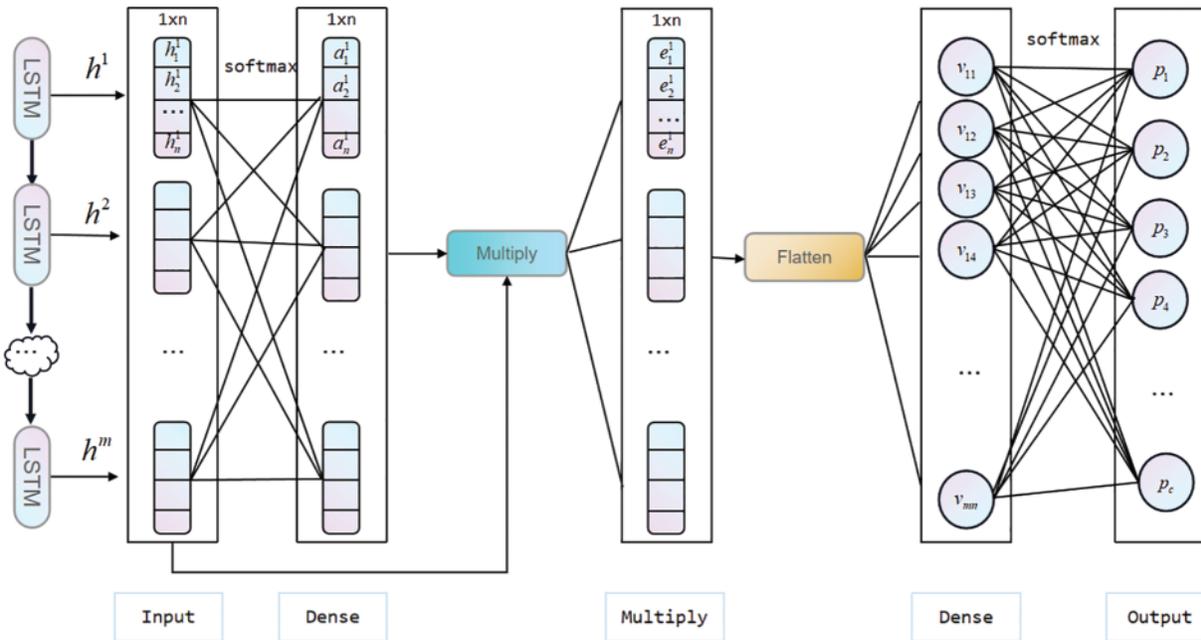


Figure 3: Attention layer structure in the proposed LREAL model. Here, h^k denotes the input vector of the k th sample containing n features

We first formulated the attention importance score, a_i^k , for each feature in the k th sample by multiplying the input vector, h_i^k , with a weight matrix, W_s , and normalizing the result to a probability

which represents the degree of relevance between the i th feature and the locomotion mode of the k th sample. It can be expressed as

$$a_i^k = \frac{\exp(s_i^k)}{\sum_{j=1}^n \exp(s_j^k)} \quad (1)$$

where $s_i^k = h_i^k W_s$.

To concentrate on the feature instead of focusing on a specific sample, each feature of a sample is assigned a separate importance score instead of a fixed score. Then we can denote the newly-constructed feature vectors of i th feature, e_i^k , as

$$e_i^k = h_i^k a_i^k \quad (2)$$

which indicates the degree of relevance between the i th feature of k th sample and the locomotion class. By flattening e_i^k , we then combine the different features of the sample within the feature vector, v_{ki} , as

$$v_{ki} = e_i^k \quad (3)$$

In conclusion, the attention layer helps the network to better predict the locomotion classes by focusing on the key sample of the analysis window.

After obtaining v_{ki} , the probability vector, $p = [p_1, p_2, \dots, p_c]$, for analysis window classification can be formulated as

$$p = \text{softmax}(w_p v_{ki} + b_p) \quad (4)$$

where w_p and b_p are the weight matrix and bias vector, respectively.

4.3 Attention-Based LSTM Design

The input of the attention-based LSTM network (Fig. 4) is an analysis window of dimension 250×12 . The three LSTM layers have 128, 64, and 32 units, respectively, and the LSTM layers will keep the structure of N-to-N, which means the output of the final LSTM layer remains a sequence. The attention layer takes the sequence as the input (Fig. 3). Finally, we add a dense layer with 100 units and a softmax layer that obtains the final classification result. The size of the softmax layer is designed based on tasks. Specifically, the layer size is set to (i) 2 when the classifier is designed to distinguish the steady and transition states, and (ii) 7/12 when recognizing the steady and transition states in detail.

4.4 Rule-Modulating Block Design

The rule-modulating block was designed based on the locomotion transformation information from the ENABL3S dataset. It combines this information with that from the steady-state classifier and passes it to the transition-state classifier for recognition. The rule was designed based on the previous locomotion mode before the transition, as we know that each steady state can only have limited options for subsequent transition states (e.g., for the S steady state, only the S-ST transition is possible; and for the ST steady state, only the ST-S and ST-LW transitions are possible). All rules were obtained based on the actual locomotion mode in ENABL3S and encoded as within the $M_{7 \times 12}$ matrix in the network (Table 2). The rows of the matrix represent the previous steady locomotion state and the columns represent the subsequent transition. If the transformation from the previous locomotion to the subsequent transition is valid(invalid), its value in the matrix is set to 1(0).

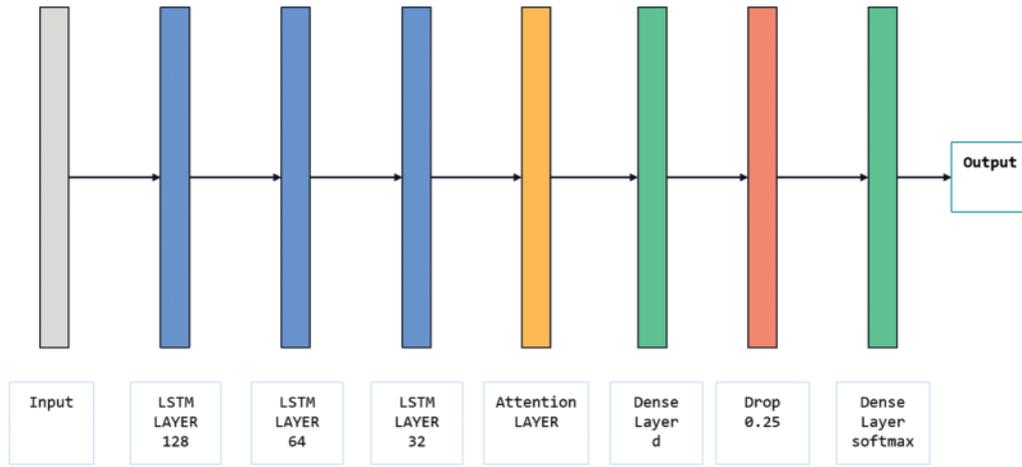


Figure 4: AT-LSTM structure of the attention-based LSTM. The dense layer size d is fixed based on the actual problem

Table 2: Locomotion mode conversion rules in the ENABL3S dataset. The rows represent the previous steady locomotion and the columns represent the subsequent transition; ‘✓’ represents valid conversions

	S-ST	ST-LW	LW-SA	SA-LW	LW-RD	RD-LW	LW-ST	ST-S	LW-RA	RA-LW	LW-SD	SD-LW
S	✓											
LW			✓		✓		✓		✓			✓
RA										✓		✓
RD						✓						
SA				✓								
SD												✓
ST		✓						✓				

After obtaining the conversion rules, the rule-modulating block combines it with the probability vector, $p = [p_1, p_2, \dots, p_7]$, of the previous analysis window from the steady-state classifier as

$$x_j = \sum_{i=1}^7 p_i \times M_{i,j} \quad j \in [1, 12] \tag{5}$$

The obtained encoded information vector $X_{1 \times 12}$, which represents the probability of the next transition, is passed on to the transition-state classifier as input. To prevent the encoded information from being diluted, it is directly input into the attention layer along with the hidden layer output vector from the previous LSTM layer to construct the final transition-state classifier (Fig. 5).

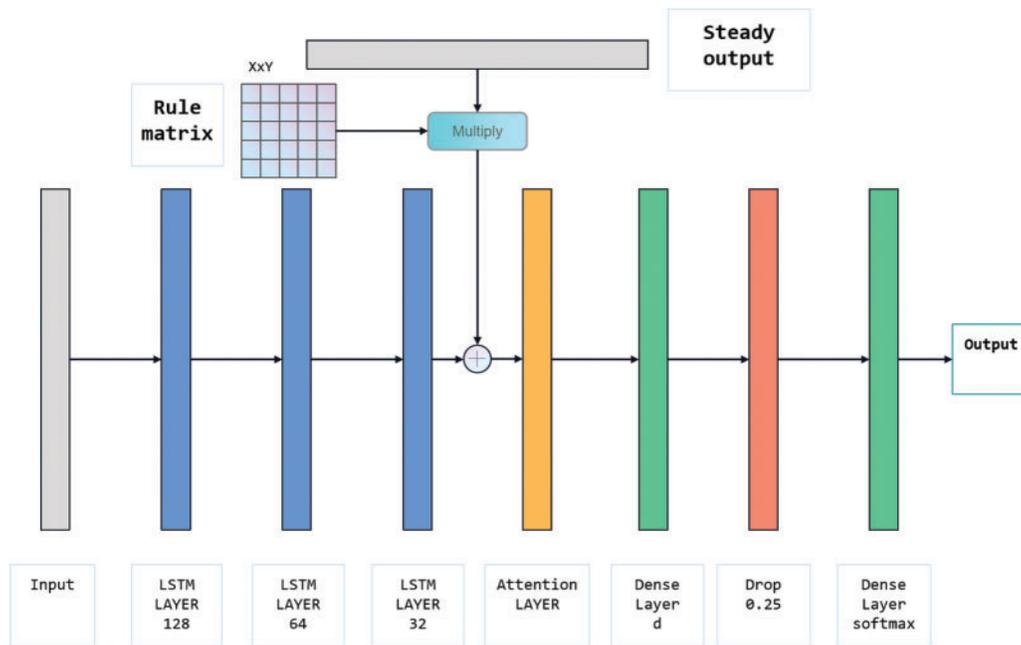


Figure 5: AT-LSTM structure for transitions

5 Experimental Setup and Discussions

In this section, we present the conducted experiments and obtained results to validate the efficacy of the proposed LREAL method. All experiments were conducted using the Keras API, with the training performed on a system with an NVIDIA RTX 3080 and AMD Ryzen 9 5900 HX processor with 32 GB RAM. We evaluated the overall network performance when handling mixed data, as well as that of each classifier in LREAL when dealing with pure steady-state or transition-state scenarios. Additionally, we compared our results with those of the existing classification methods to demonstrate its accuracy.

5.1 Experimental Setup

We utilized an initial set of hyperparameters and investigated their impact on the model to narrow down the range for the subsequent grid search (Table 3). A grid search with a 5-fold cross-validation was then performed on the hyperparameters to maximize validation accuracy (Table 4; a detailed discussion on the influence of the hyperparameters can be found in Section 5.3.6). The configuration shown in Table 4 was selected as the optimal choice for subsequent experiments involving the classifier. To deal with the imbalanced distribution of the different classes in the ENABL3S dataset (Table 1), we adjusted the weight of each class for training inversely proportional to class frequency in the dataset as

$$\text{Weighting} = \frac{n_{\text{samples_with_class}}}{n_{\text{classes}} * n_{\text{samples_with_class}}} \quad (6)$$

To validate the effectiveness of the proposed method, we devised two distinct experimental scenarios. The first scenario aimed to evaluate the classification performance of each LREAL component individually. Thus, the performance of the three classifiers was tested separately in classifying pure data for steady/transitions states. In the second scenario, we introduced mixed data and examined the

performance of LREAL when considering the mutual influence of various components. Furthermore, in both scenarios, the networks were trained and tested on user-dependent and user-independent base.

Table 3: Hyperparameters values

Parameter	Value
Epoch	200
Learning rate	0.0001
Batch size	256
Optimizer	Adam [33]
Loss function	Cross-entropy
Class-weighting	True
Early stopping	10 epochs if no improvement on validation loss

Table 4: Grid search results

Classifier	Learning rate	Optimizer	Batch size
Steady/transition	0.001	Adam	64
Steady state	0.001	Adam	64
Transition	0.0005	RMSprop	64

5.2 Performance Evaluation

We used the accuracy (ACC) score to measure the overall performance of our proposed LREAL network; it can be expressed as

$$ACC = \frac{N_{correct}}{N_{total}} \times 100\% \quad (7)$$

where $N_{correct}$ is the number of correct classifications and N_{total} is the total number of data points. We also evaluated the network performance in two bases: (1) Data from one out of ten subjects was used to train and test a user-dependent network; (2) data from nine out of ten subjects was used as the training set, while the remaining subject was used as test set to construct a user-independent network. The processes were repeated 10 times until all subjects were used.

Furthermore, when evaluating the overall performance, the number of sensors utilized serves as a crucial metric. In terms of practical applications, achieving comparable results with fewer sensors signifies superior model performance.

5.3 Discussion

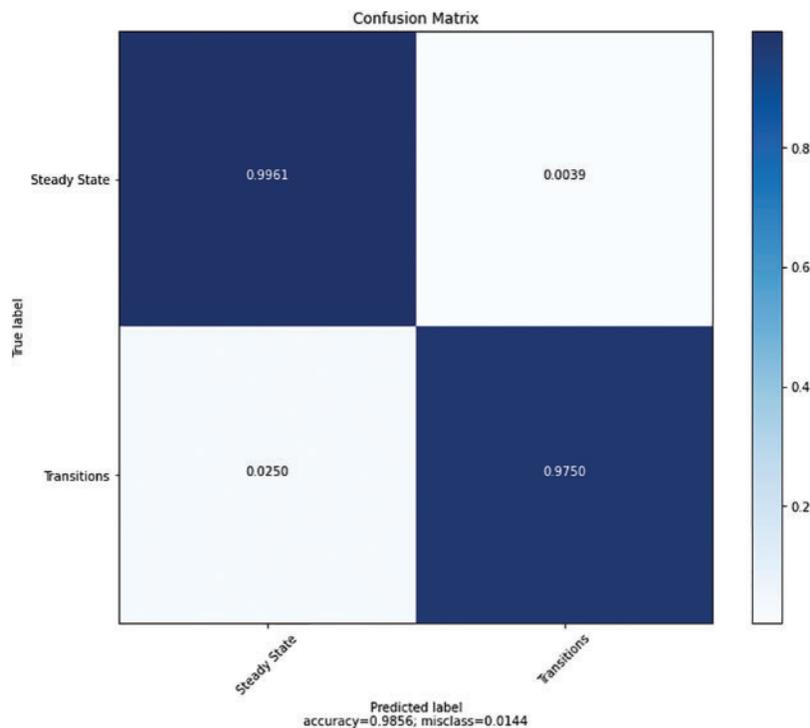
5.3.1 Single Classifier Evaluation

To evaluate and analyze the performance of the three types of classifiers in the proposed network (Fig. 2), we computed their average accuracies on the user-dependent and user-independent bases. The results of individual classifiers did not consider their mutual influence to evaluate their actual capacity for recognizing locomotor intent (Table 5).

Table 5: Performance of the three types of LREAL classifiers

Recognition classifier	User-dependent ACC	User-independent
	ACC (%)	ACC (%)
Steady/transition	98.56 ± 0.35	96.36 ± 2.12
Steady state	99.32 ± 0.18	96.18 ± 2.25
Transition	98.96 ± 0.32	92.51 ± 3.16

The first-level steady/transition state classifier achieved an accuracy of (98.56% ± 0.35%) and (96.42% ± 2.12%) on the user-dependent and user-independent bases, respectively. A confusion matrix is a table that visualizes the performance of a classifier, comprising data of the true and predicted labels that the model evaluated. We obtained and analyzed the confusion matrix of the user-dependent classifier (Fig. 6). Transition class obtained an accuracy of 97.50% much lower than that for the steady state class (99.61%). A previous study has shown that steady RA and RD locomotion can introduce a high error rate for the similarities between ramp and ground-level walking, as well as the transitions between these steady states [32]. The performance of the steady/transition classifier largely affects the end-stage classification accuracy and the overall network performance.

**Figure 6:** Confusion matrix for the steady/transition classifier on the user-dependent

A previous study developed a recognition system based on a Gaussian SVM and kinematic data which was able to classify five steady states and eight transition states [34], and the steady/transition classifier in it had an accuracy of 93.30%. Our framework was more accurate (+5.26%) while

our classifiers utilized the kinematic signals with the data from GONIO sensors. Although our steady/transition classifier demonstrated higher accuracy, further analysis is required to improve its effectiveness.

The steady-state classifier was effective on both user-independent ($ACC = 99.32\%$) and user-dependent ($ACC = 96.18\%$) bases. However, the transitions-state classifier was more accurate in the user-independent ($ACC = 98.96\%$) basis compared to the user-dependent ($ACC = 92.51\%$) basis. The errors were mainly due to the transitions between RD/RA and LW. The differences in individual exercise methods and the complexity of transitions are expected to explain this observation.

We also compared the user-dependent classifier with the existing method when facing pure transition or steady states data (Tables 6 and 7). Although our approach was slightly less effective in recognizing steady states compared to a multi-level SVM [34], it was more versatile (recognized seven types of steady states compared to the five types identified by the SVM) with similar efficacy. Furthermore, our approach considered more transition states and achieved a much higher accuracy than the multi-level SVM (+3.06%). Compared to context-based Bayesian [35], central pattern generator (CPG) [36], and CNN-LSTM networks [37], our AT-LSTM network surpassed in terms of accuracy for both steady and transition-state recognition when considering more classes. These results validated the effectiveness of AT-LSTM in recognizing human locomotor intent.

Table 6: Comparison of various methods for steady-state recognition

Method	Number of recognizable classes	Accuracy (%)
SVM (multi-level) [34]	5	99.4
Context-based Bayesian [35]	7	98.3
CPG [36]	5	99.03
CNN-LSTM [37]	7	99.1
AT-LSTM	7	99.32

Table 7: Comparison of various methods for transition-state recognition

Method	Number of recognizable classes	Accuracy (%)
SVM (multi-level) [34]	8	95.9
Context-based Bayesian [35]	12	98.8
CPG [36]	8	98.9
CNN-LSTM [37]	12	96.9
AT-LSTM	12	98.96

5.3.2 Overall Network Evaluation

The overall performance of the LREAL network was further evaluated by calculating the recognition accuracy across the entire network, i.e., considering the impact of all classifiers. The network achieves an average accuracy of 98.60% and 92.78% on the user-dependent and user-independent base, respectively. Additionally, the user-dependent (user-independent) network obtained an average accuracy of 99.03% (94.25%) and 96.52% (86.52%) for the steady and transition states,

respectively. From the confusion matrix of the user-dependent LREAL network (Fig. 7), it can be seen that the errors mainly belong to the recognition of transitions, including misclassification between transitions and between transition and steady states. Certain steady states were also misclassified as transition states. This can be attributed to the similarity between transition and steady states and partial locomotion.

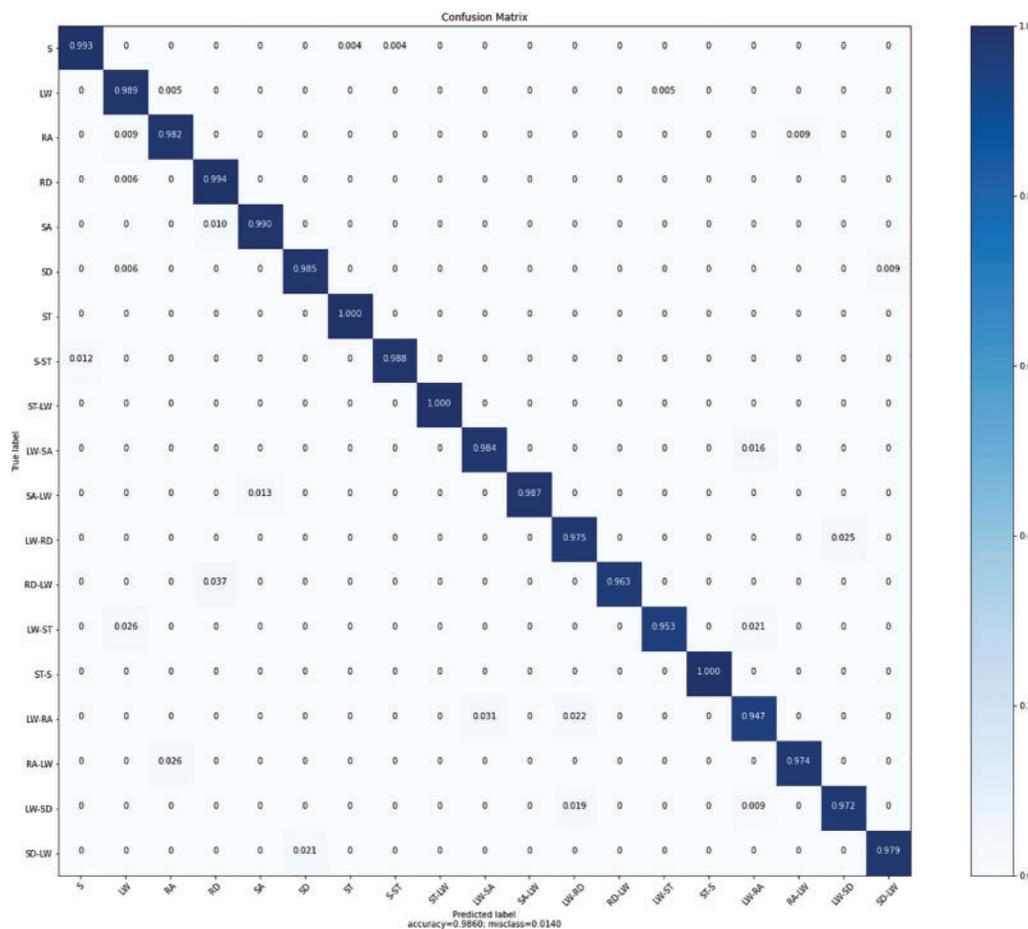


Figure 7: Confusion matrix for the user-dependent LREAL network

On the other hand, it is worth noting that the scope of misclassification is narrow, as the rule-embedded information restrains the classification range of each class in the case of correct first level classification. However, our approach took a longer time to recognize locomotor intents as there were three classifiers working together. The average running time for the recognition of one sequence when classifying 10,000 sequences was (112.06 ± 6.23) ms for both user-dependent and user-independent base.

Furthermore, we compared the performance of the LREAL network with previous methods using ENABL3S according to the recognizable classes, the number of used sensors, and accuracy (Table 8). The results demonstrated that the LREAL network outperformed LDA [8], CNN-LSTM [37], CNN [38], and Light-Weight Artificial Neural Network (LWANN) [39] in terms of accuracy for steady-state recognition, and surpassed LIR-Net [27], LDA, CNN, LWANN, and CNN-LSTM in terms of

accuracy for transition recognition. While LIR-Net achieves better overall performance and accuracy for steady-state recognition compared to LREAL, it only recognizes five steady states and eight types of transitions compared to the seven steady states and twelve types of transitions recognized by the LREAL network. Additionally, LIR-Net utilizes a larger number of sensors compared to LREAL. LREAL demonstrated superior performance in handling transitions, albeit with a slightly lower overall accuracy than that of LIR-Net due to fewer sensors (two IMU and four GONIO) and a larger number of recognizable locomotor intent classes. In summary, LREAL used fewer sensors to achieve a higher accuracy for a higher number of classes.

Table 8: Performance comparison of the LREAL network with other methods on ENABL3S

Method	Sensors	Classes		Accuracy (%)		
		Number of steady states	Number of transitions	Steady states	Transitions	Overall
LIR-Net [27]	14 EMG, 4 IMU, 4 GONIO	5	8	99.46	96.34	98.89
LDA [8]	14 EMG, 4 IMU, 4 GONIO	7	12	98.75	94.06	98.48
CNN [38]	14 EMG, 5IMU 4 GONIO	5	8	NULL	NULL	96.3
LWANN [39]	5 IMU 4GONIO	7	12	NULL	NULL	98.60
CNN-LSTM [37]	2 IMU	7	12	96.41	94.32	95.82
LREAL	2 IMU 4GONIO	7	12	99.03	96.52	98.60

5.3.3 AM Visualization

To better understand and explain the internal operations of the AM the locomotor intent recognition task, we visualized its performance from two aspects.

Firstly, we analyzed the performance of AM on the features of a sample. We extracted the activations of the attention layer of the steady-state classifier as an example. Let us denote the simple average of hidden vectors from the previous LSTM layer as h_i and their weighted sum after the attention layer as h_i^a . To make the visualization clearer, we normalized the activations after obtaining them. In Fig. 8, each column represents a feature dimension, and the rows correspond to the feature vectors before and after the action of the attention layer. The attention layer modifies h_i^a by redistributing the weight of each feature.

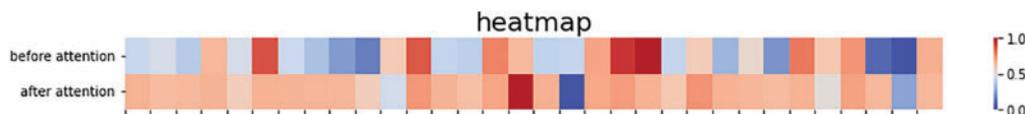


Figure 8: Visualization of feature vectors before and after the action of the attention layer

We also analyzed the ability of the AM to focus on key instances of the time sequence for the steady states and transitions. Specifically, using the attention importance score for each feature, a_i^k , in the k th sample, we calculated the average importance score, a^k , for the k th sample to study the preference

of the AM for the overall analysis window. The attention layer exhibits different preferences for the tasks of steady and transition-state recognition (Figs. 9a and 9b). The attention layer focused more on the intermediate fragment of the analysis window for steady-state recognition (Fig. 9a), whereas it concentrates on both ends of the analysis window for transitions (Fig. 9b). This indicated that the information that helps identify steady states is concentrated in the middle of the analysis window and the signals that dictate transitions are located on either of its end segments. Although previous studies reached similar findings for transitions by visualizing the activations of CNNs [40], they considered only the activations for transitions. Our work showed relied on the importance scores of both steady and transition states, which validates the capacity of the AM to capture valuable temporal information when facing different tasks. The finding is also critical for recognizing locomotor intent and is worth further study.

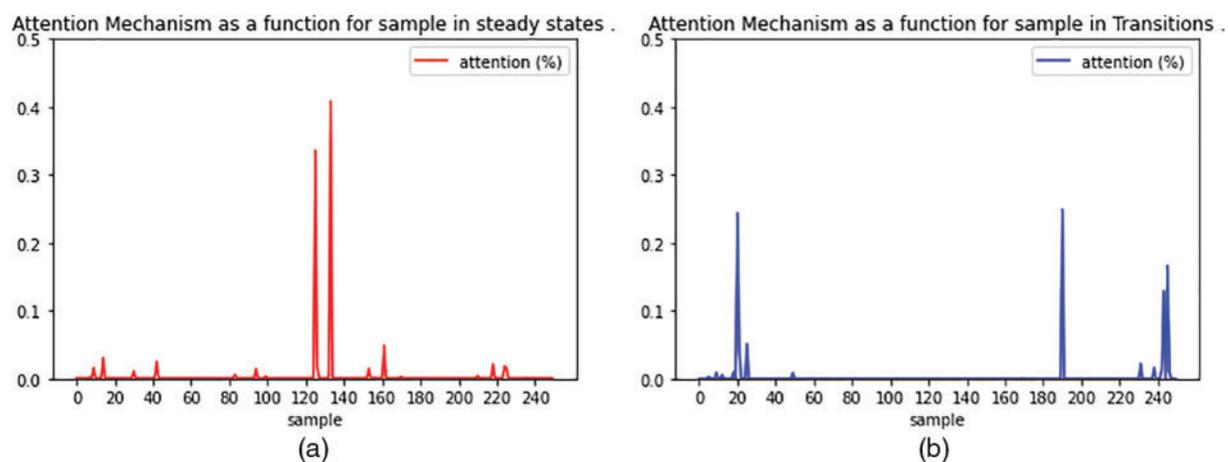


Figure 9: Relationship of the attention weights of each sample over the complete analysis window for (a) steady-state and (b) transition recognition

5.3.4 Impact of a Priori Knowledge

To analyze the impact of a priori knowledge in LREAL, we conducted tests without a priori knowledge using pure transition data. The resulting confusion matrix clearly illustrated the higher tendency for misrecognition of reversed transition states without a priori knowledge (Fig. 10). For instance, the LW-RA transitions were easily misclassified as RA-LW transitions and vice versa. The incorporation of a priori knowledge imposed constraints on the classifiers and reduced the occurrence of such misjudgments by enabling the model to learn from previous states. It enabled the model to gain a better understanding of movement rules, leading to improved performance.

5.3.5 Ablation Study

To investigate the effectiveness of each component in the LREAL network, we conducted ablation analyses on its second level (constructed by incorporating the AM into LSTM and connecting the two classifiers with a rule-modulating block) based on the ENABL3S dataset. We compared the results with the following model:

1. Raw LSTM: Two LSTM classifiers without an attention layer or a rule-modulating block between them.

2. Rule-Embedded (RE)-LSTM: Two LSTM classifiers with a rule-modulating block to deliver rule-embedded knowledge between them.
3. AT-LSTM: Two classifiers constructed using AT-LSTM, but without a rule-modulating block to input locomotion knowledge between them.

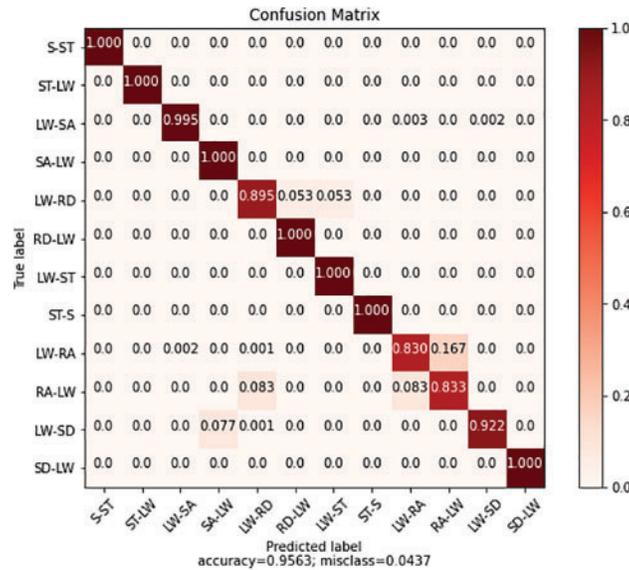


Figure 10: Confusion matrix for LREAL without a priori knowledge when facing transitions

From the ablation study results (Table 9), we observed that the addition of a rule-modulating block greatly improved the accuracy of recognizing transitions compared to the network without it. In summary, LREAL indeed outperformed all other variants.

Table 9: Ablation study comparison for LREAL with other variant models

Method	Accuracy (%)	
	Steady states	Transitions
Raw LSTM	94.16	92.22
RE-LSTM	96.88	98.37
AT-LSTM	99.32	95.63
LREAL	99.32	98.96

5.3.6 Parameter Analysis

To achieve the best performance for the hybrid network, the choice of parameters should be considered carefully and the parameters of each sub-classifier must be tuned and optimized. To assess the influence of the different hyperparameters (Table 3) on the model and streamline the grid search process, a series of ablation experiments were conducted on the individual sub-classifiers and the hybrid network.

We conducted ablation experiments by considering the significant impact of the learning rate and optimizer on classifier performance, and recognizing that each optimizer has its own optimal learning rate. Specifically, we evaluated three optimizers, namely Adam, stochastic gradient descent (SGD) with momentum, and root mean squared propagation (RMSprop). Each optimizer was tested for a set of learning rates (0.01, 0.005, 0.001, 0.0005, and 0.0001). Analyzing the results (Fig. 11), we observed that for the steady-state classifier and steady/transition classifier, the Adam optimizer with a learning rate of 0.001 outperformed the other combinations. However, for the transition-state classifier, the RMSprop optimizer with a learning rate of 0.0005 achieved the best performance compared to the other combinations (Fig. 11).

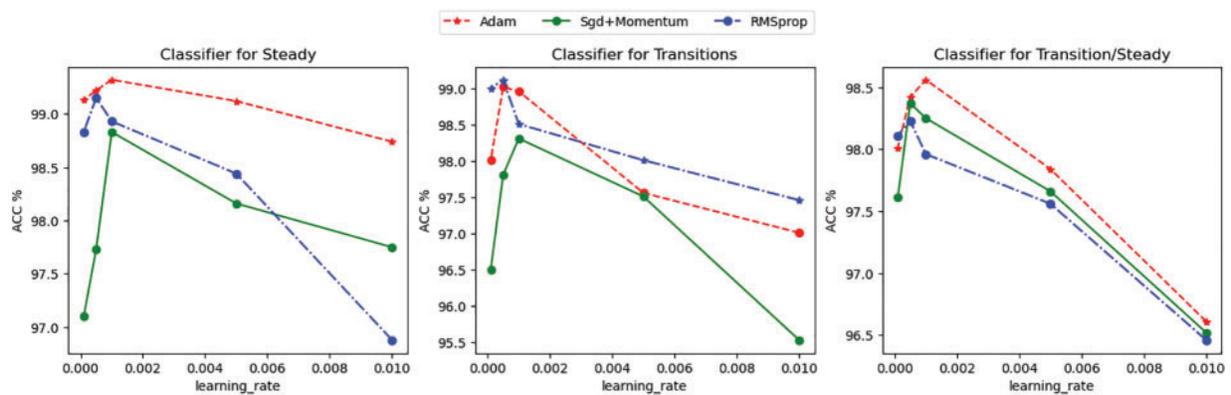


Figure 11: Impact of optimizers and learning rates on the sub-classifier performance

Finding a suitable batch size is crucial as it can effectively reduce memory usage and enhance the model generalize ability. We evaluated the impact of different batch sizes (32, 64, 128, 256, and 512) on the overall performance of the network (Fig. 12). Our findings indicated that LREAL achieved the best overall effect for a batch size of 64.

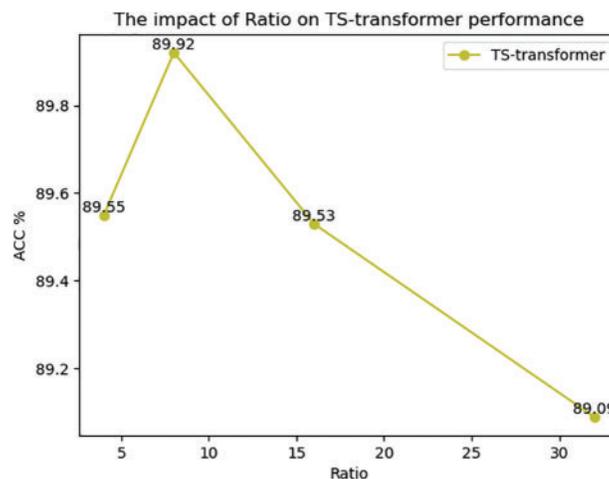


Figure 12: Effect of batch size on LREAL performance

5.3.7 Limitations and Future Outlook

- 1) *Rule-modulating block design*: Although we designed a simple rule to incorporate into the network, its design of rule is based on the specific locomotor circuit in the ENABL3S dataset. When the situation becomes much more complex, e.g., considering more daily locomotion the rule will become diverse and the rule-embedded module may not be very effective. The method of combining a rule-based system with DL remains relatively simple. Therefore, a more comprehensive study of such combinations may improve the impact of rule-embedded networks and model interpretability.
- 2) *First-level classifier accuracy*: The performance of the first-level steady/transition classifier largely affected the overall network effectiveness as its classification errors propagate to the final classification. The steady/transition classifier achieved an average accuracy of 97.50% when facing transition states (Fig. 6). This might be due to the manner of extracting the analysis window containing steady-state information before the transition. The accuracy may be improved by a further study of extracting transition-state analysis windows. Moreover, the unbalanced distribution of data may also contribute to this issue. A balanced dataset containing more transition state data could enhance the classifier performance.
- 3) *Practical application requirements*: To meet the needs of practical applications and reduce user discomfort, the latency of recognition should be <300 ms [41]; our method can currently classify a sequence in 112 ms. However, when considering the 50 ms sliding window for sequence extraction and the time taken to preprocess the data, the total recognition time increases greatly (>300 ms). As for a microcomputer on a real prosthesis, it might take a longer time to process data and recognize locomotor intent. Therefore, further research is required to reduce the system running time to meet the needs of practical applications.

6 Conclusion

In this study, we propose a locomotion rule embedding LSTM network with an AM (LREAL) based on IMU and GONIO sensors to recognize human locomotor intent. The performance of the overall network and each classifier was validated on the public ENABL3S dataset and compared to other methods. The results show that the locomotion recognition of seven steady and twelve transition states achieved an average accuracy of 99.03% and 96.52%, respectively, on the ENABL3S dataset, i.e., comparable to or better than the other methods. We also analyzed the impact of the AM from two aspects and verified the effectiveness of adding the rule-modulating block by ablation studies. To summarize, this study demonstrates the excellent human locomotor intent recognition capabilities of AM and hence, the proposed LREAL network, which is important for the further study and development of assistive robotic.

Acknowledgement: We would like to express our gratitude to Haoming Da for his invaluable assistance with the experiments, and to Hui Yang for her significant contributions to language polishing and editing. Their expertise and insights have greatly enhanced the quality of this work.

Funding Statement: This research was funded by the National Natural Science Foundation of China (Nos. 62072212, 62302218), the Development Project of Jilin Province of China (Nos. 20220508125RC, 20230201065GX, 20240101364JC), National Key R&D Program (No. 2018YFC2001302), and the Jilin Provincial Key Laboratory of Big Data Intelligent Cognition (No. 20210504003GH).

Author Contributions: The authors confirm contribution to the paper as follows: Software, writing—original draft preparation, writing—review and editing: J. Shen; formal analysis, project administration and supervision: Y. Wang; investigation and validation: D. Zhang. All authors reviewed the results and approved the final version of the manuscript.

Availability of Data and Materials: We using an open-source dataset to test our method. The ENABL3S dataset can be found in https://figshare.com/articles/dataset/Benchmark_datasets_for_bilateral_lower_limb_neuromechanical_signals_from_wearable_sensors_during_unassisted_locomotion_in_able-bodied_individuals/5362627 (accessed on 18 December 2021).

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

- [1] J. Verghese, A. LeValley, C. B. Hall, M. J. Katz, A. F. Ambrose and R. B. Lipton, “Epidemiology of gait disorders in community-residing older adults,” *J. Am. Geriatr. Soc.*, vol. 54, no. 2, pp. 255–261, 2006. doi: [10.1111/j.1532-5415.2005.00580.x](https://doi.org/10.1111/j.1532-5415.2005.00580.x).
- [2] H. Huang, F. Zhang, L. J. Hargrove, Z. Dou, D. R. Rogers and K. B. Englehart, “Continuous locomotion-mode identification for prosthetic legs based on neuromuscular-mechanical fusion,” *IEEE Trans. BioMed. Eng.*, vol. 58, no. 10, pp. 2867–2875, Oct. 2011. doi: [10.1109/TBME.2011.2161671](https://doi.org/10.1109/TBME.2011.2161671).
- [3] Y. X. Liu and E. M. Gutierrez-Farewik, “Joint kinematics, kinetics and muscle synergy patterns during transitions between locomotion modes,” *IEEE Trans. Biomed. Eng.*, vol. 70, no. 3, pp. 1062–1071, Mar. 2023. doi: [10.1109/TBME.2022.3208381](https://doi.org/10.1109/TBME.2022.3208381).
- [4] H. Lu, L. R. B. Schomaker, and R. Carloni, “IMU-based deep neural networks for locomotor intention prediction,” in *2020 IEEE/RSJ Int. Conf. Intell. Robot. Syst. (IROS)*, Las Vegas, NV, USA, 2020, pp. 4134–4139.
- [5] A. Mai and S. Commuri, “Gait identification for an intelligent prosthetic foot,” in *2011 IEEE Int. Symp. Intell. Control*, Denver, CO, USA, 2011, pp. 1341–1346.
- [6] Q. Yu, P. Zhang, and Y. Chen, “Human motion state recognition based on flexible, wearable capacitive pressure sensors,” *Micromachines*, vol. 12, no. 10, pp. 1219, Oct. 2021. doi: [10.3390/mi12101219](https://doi.org/10.3390/mi12101219).
- [7] J. Camargo, W. Flanagan, N. Csomay-Shanklin, B. Kanwar, and A. Young, “A machine learning strategy for locomotion classification and parameter estimation using fusion of wearable sensors,” *IEEE Trans. Biomed. Eng.*, vol. 68, no. 5, pp. 1569–1578, May 2021. doi: [10.1109/TBME.2021.3065809](https://doi.org/10.1109/TBME.2021.3065809).
- [8] B. Hu, E. Rouse, and L. Hargrove, “Fusion of bilateral lower-limb neuromechanical signals improves prediction of locomotor activities,” *Front. Rob. AI*, vol. 5, pp. 78–78, Jun. 2018. doi: [10.3389/frobt.2018.00078](https://doi.org/10.3389/frobt.2018.00078).
- [9] S. Xu and Y. Ding, “Real-time recognition of human lower-limb locomotion based on exponential coordinates of relative rotations,” *Sci. China Technol. Sci.*, vol. 64, pp. 1423–1435, Jul. 2021. doi: [10.1007/s11431-020-1802-2](https://doi.org/10.1007/s11431-020-1802-2).
- [10] D. Xu and Q. Wang, “On-board training strategy for IMU-based real-time locomotion recognition of transtibial amputees with robotic prostheses,” *Front. Neurobot.*, vol. 14, no. 47, pp. 608, Oct. 2020. doi: [10.3389/fnbot.2020.00047](https://doi.org/10.3389/fnbot.2020.00047).
- [11] D. Xu, Y. Feng, J. Mai, and Q. Wang, “Real-time on-board recognition of continuous locomotion modes for amputees with robotic transtibial prostheses,” *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 26, no. 10, pp. 2015–2025, Oct. 2018. doi: [10.1109/TNSRE.2018.2870152](https://doi.org/10.1109/TNSRE.2018.2870152).
- [12] D. H. Moon, D. Kim, and Y. D. Hong, “Development of a single leg knee exoskeleton and sensing knee center of rotation change for intention detection,” *Sensors*, vol. 19, no. 18, pp. 3960, 2019. doi: [10.3390/s19183960](https://doi.org/10.3390/s19183960).

- [13] Z. Lu, A. Narayan, and H. Yu, "A deep learning based end-to-end locomotion mode detection method for lower limb wearable robot control," in *2020 IEEE/RSJ Int. Conf. Intell. Robot. Syst. (IROS)*, Las Vegas, NV, USA, 2020, pp. 4091–4097.
- [14] B. Hu, P. C. Dixon, J. V. Jacobs, J. T. Dennerlein, and J. M. Schiffman, "Machine learning algorithms based on signals from a single wearable inertial sensor can detect surface- and age-related differences in walking," *J. Biomech.*, vol. 71, pp. 37–42, Apr. 2018. doi: [10.1016/j.jbiomech.2018.01.005](https://doi.org/10.1016/j.jbiomech.2018.01.005).
- [15] J. Bruinsma and R. Carloni, "IMU-based deep neural networks: Prediction of locomotor and transition intentions of an osseointegrated transfemoral amputee," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 29, pp. 1079–1088, 2021. doi: [10.1109/TNSRE.2021.3086843](https://doi.org/10.1109/TNSRE.2021.3086843).
- [16] K. Arulkumaran, M. P. Deisenroth, M. Brundage, and A. A. Bharath, "Deep reinforcement learning: A brief survey," *IEEE Signal Process. Mag.*, vol. 34, no. 6, pp. 26–38, Nov. 2017. doi: [10.1109/MSP.2017.2743240](https://doi.org/10.1109/MSP.2017.2743240).
- [17] V. François-Lavet, P. Henderson, R. Islam, M. G. Bellemare, and J. Pineau, "An introduction to deep reinforcement learning," *Found Trends Mach. Learn.*, vol. 11, no. 3–4, pp. 219–354, Dec. 2018. doi: [10.1561/22000000071](https://doi.org/10.1561/22000000071).
- [18] J. Ahn, J. Lee, and L. Sentis, "Data-efficient and safe learning for humanoid locomotion aided by a dynamic balancing model," *IEEE Robot. Autom. Lett.*, vol. 5, no. 3, pp. 4376–4383, Jul. 2020. doi: [10.1109/LRA.2020.2990743](https://doi.org/10.1109/LRA.2020.2990743).
- [19] R. Zheng, Z. Yu, H. Liu, J. Chen, Z. Zhao and L. Jia, "End-to-end high-level control of lower-limb exoskeleton for human performance augmentation based on deep reinforcement learning," *IEEE Access*, vol. 11, pp. 102340–102351, 2023. doi: [10.1109/ACCESS.2023.3317183](https://doi.org/10.1109/ACCESS.2023.3317183).
- [20] X. Zhou, W. Liang, K. I. K. Wang, H. Wang, L. T. Yang and Q. Jin, "Deep-learning-enhanced human activity recognition for internet of healthcare things," *IEEE Internet Things J.*, vol. 7, no. 7, pp. 6429–6438, July 2020. doi: [10.1109/JIOT.2020.2985082](https://doi.org/10.1109/JIOT.2020.2985082).
- [21] C. Wang, W. Guo, H. Zhang, L. Guo, C. Huang and C. Lin, "sEMG-based continuous estimation of grasp movements by long-short term memory network," *Biomed. Signal Process. Control*, vol. 59, pp. 101774, May 2020. doi: [10.1016/j.bspc.2019.101774](https://doi.org/10.1016/j.bspc.2019.101774).
- [22] M. Guo *et al.*, "Attention mechanisms in computer vision: A survey," *Comput. Vis. Media*, vol. 8, no. 3, pp. 331–368, Sep. 2022. doi: [10.1007/s41095-022-0271-y](https://doi.org/10.1007/s41095-022-0271-y).
- [23] C. Zhu, Q. Liu, W. Meng, Q. Ai, and S. Xie, "An attention-based CNN-LSTM model with limb synergy for joint angles prediction," in *2021 IEEE/ASME Int. Conf. Adv. Intell. Mechatron. (AIM)*, Delft, Netherlands, 2021, pp. 747–752.
- [24] H. A. Varol, F. Sup, and M. Goldfarb, "Multiclass real-time intent recognition of a powered lower limb prosthesis," *IEEE Trans. Biomed. Eng.*, vol. 57, no. 3, pp. 542–551, Mar. 2010. doi: [10.1109/TBME.2009.2034734](https://doi.org/10.1109/TBME.2009.2034734).
- [25] H. Wang, "ReNN: Rule-embedded neural networks," in *2018 24th Int. Conf. Pattern Recogn. (ICPR)*, Beijing, China, 2018, pp. 824–829.
- [26] B. Hu, E. Rouse, and L. Hargrove, "Benchmark datasets for bilateral lower-limb neuromechanical signals from wearable sensors during unassisted locomotion in able-bodied individuals," *Front. Rob. AI*, vol. 5, pp. 14, Feb. 2018. doi: [10.3389/frobt.2018.00014](https://doi.org/10.3389/frobt.2018.00014).
- [27] U. H. Lee, J. Bi, R. Patel, D. Fouhey, and E. Rouse, "Image transformation and CNNs: A strategy for encoding human locomotor intent for autonomous wearable robots," *IEEE Robot. Autom. Lett.*, vol. 5, no. 4, pp. 5440–5447, Oct. 2020. doi: [10.1109/LRA.2020.3007455](https://doi.org/10.1109/LRA.2020.3007455).
- [28] H. Wang *et al.*, "Wearable sensor-based human activity recognition using hybrid deep learning techniques," *Secur. Commun. Netw.*, vol. 2020, pp. 1–12, Jul. 2020. doi: [10.1155/2020/2132138](https://doi.org/10.1155/2020/2132138).
- [29] Z. Zheng, L. Shi, C. Wang, L. Sun, and G. Pan, "LSTM with uniqueness attention for human activity recognition," in *Artificial Neural Networks and Machine Learning–ICANN 2019*, Munich, Germany, 2019, pp. 498–509.
- [30] C. Zhang, S. Bengio, M. Hardt, B. Recht, and O. Vinyals, "Understanding deep learning (still) requires rethinking generalization," *Commun. ACM.*, vol. 64, pp. 107–115, 2021. doi: [10.1145/3446776](https://doi.org/10.1145/3446776).

- [31] G. Winter, “*xia2*: An expert system for macromolecular crystallography data reduction,” *J. Appl. Crystallogr.*, vol. 43, pp. 186–190, Feb. 2010. doi: [10.1107/S0021889809045701](https://doi.org/10.1107/S0021889809045701).
- [32] A. J. Young and L. J. Hargrove, “A classification method for user-independent intent recognition for transfemoral amputees using powered lower limb prostheses,” *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 24, no. 2, pp. 217–225, Feb. 2016. doi: [10.1109/TNSRE.2015.2412461](https://doi.org/10.1109/TNSRE.2015.2412461).
- [33] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” in *3rd Int. Conf. Learn. Represent. (ICLR)*, San Diego, USA, 2015.
- [34] J. Figueiredo, S. P. Carvalho, D. Goncalve, J. C. Moreno, and C. P. Santos, “Daily locomotion recognition and prediction: A kinematic data-based machine learning approach,” *IEEE Access*, vol. 8, pp. 33250–33262, 2020. doi: [10.1109/ACCESS.2020.2971552](https://doi.org/10.1109/ACCESS.2020.2971552).
- [35] U. Martinez-Hernandez, L. Meng, D. Zhang, and A. Rubio-Solis, “Towards a context-based Bayesian recognition of transitions in locomotion activities,” in *2020 29th IEEE Int. Conf. Robot Hum. Interact. Commun. (RO-MAN)*, Naples, Italy, 2020, pp. 677–682.
- [36] Y. Wang, X. Cheng, L. Jabban, X. Sui, and D. Zhang, “Motion intention prediction and joint trajectories generation toward lower limb prostheses using EMG and IMU signals,” *IEEE Sens. J.*, vol. 22, no. 11, pp. 10719–10729, Jun. 2022. doi: [10.1109/JSEN.2022.3167686](https://doi.org/10.1109/JSEN.2022.3167686).
- [37] D. Marcos Mazon, M. Groefsema, L. R. Schomaker, and R. Carloni, “IMU-based classification of locomotion modes, transitions, and gait phases with convolutional recurrent neural networks,” *Sensors*, vol. 22, no. 22, pp. 8871, 2022. doi: [10.3390/s22228871](https://doi.org/10.3390/s22228871).
- [38] K. Zhang, J. Wang, C. W. de Silva, and C. Fu, “Unsupervised cross-subject adaptation for predicting human locomotion intent,” *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 28, no. 3, pp. 646–657, Mar. 2020. doi: [10.1109/TNSRE.2020.2966749](https://doi.org/10.1109/TNSRE.2020.2966749).
- [39] S. A. Mohamed and U. Martinez-Hernandez, “A light-weight artificial neural network for recognition of activities of daily living,” *Sensors*, vol. 23, no. 13, pp. 5854, Jul. 2023. doi: [10.3390/s23135854](https://doi.org/10.3390/s23135854).
- [40] B. Y. Su *et al.*, “A CNN-based method for intent recognition using inertial measurement units and intelligent lower limb prosthesis,” *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 27, no. 5, pp. 1032–1042, May 2019. doi: [10.1109/TNSRE.2019.2909585](https://doi.org/10.1109/TNSRE.2019.2909585).
- [41] K. Englehart and B. Hudgins, “A robust, real-time control scheme for multifunction myoelectric control,” *IEEE Trans. Biomed. Eng.*, vol. 50, no. 7, pp. 848–854, July 2003. doi: [10.1109/TBME.2003.813539](https://doi.org/10.1109/TBME.2003.813539).