Deep Learning-Based Torque Estimation for Human-Robot Interaction: Experimental Setup and Preliminary Results

Eric Ribeiro Alves, Lucas Martins Morello, Rafael Baldessini Campanini

[a] *Department of Control and Automation Engineering, UNESP, Av. 3 de Março, 511, Sorocaba, SP, Brazil*

Abstract

Human-robot interaction plays a fundamental role in rehabilitation robotics, especially in devices such as exoskeletons aimed at supporting lower limbs. A critical challenge in this context is developing control systems capable of adapting to the various impedances of human joints, ensuring a smooth and safe response during motor assistance. This work proposes a Long Short-Term Memory (LSTM) network-based approach for real-time interaction torque estimation, aiming to enhance the transparency and efficiency of the exoskeleton. The methodology includes data collection via sensors, preprocessing, modeling with recurrent neural networks, and experimental evaluation. Initial results demonstrate the feasibility of the approach, with metrics indicating a reduction in prediction error compared to traditional control models.

Keywords: Deep Learning, Human-Robot Interaction, Torque Estimation, Transparency, Exoskeletons

1. Introduction

Human-robot interaction has become increasingly relevant in the development of rehabilitation technologies, particularly exoskeletons designed for assistance with lower limbs. These devices are used to facilitate the mobility of people with disabilities, providing mechanical support that must adjust to the variability of movement and resistance of the user. One of the main challenges in this scenario is adapting control for different mechanical impedances, that is, the passive resistance of human joints in response to external forces. Efficiently controlling this interaction requires an accurate estimation of interaction torque to ensure that assisted movement occurs naturally and safely.

With the advancement of deep learning techniques, neural networks, especially Long Short-Term Memory (LSTM) networks, emerge as potential solutions for predicting in real-time the torque needed for assisted movement. LSTMs are well-suited for this type of task due to their ability to capture long-term temporal dependencies, making them applicable in the context of adaptive control in robotics. This article presents an LSTM-based approach for estimating interaction torque in a rehabilitation exoskeleton, exploring how this technique can optimize the device's response to variations in human movement.

The structure of this article is as follows: In Section 2, we discuss related works and the state of the art in adaptive control for rehabilitation robotics; in Section 3, we present the proposed approach, detailing the LSTM network architecture; in Section 4, we describe the experiments and the results obtained; finally, Section 5 presents the conclusions and suggestions for future work.

2. Related Works

In recent years, various approaches have been explored for the control of exoskeletons and rehabilitation devices. The main techniques involve robust control methods, Kalman filters, and more recently, machine learning. A literature review focused on works from the last five years reveals two major strands:

1. **Robust Control-Based Methods**: Traditional methods such as torque-assisted control and impedance compensation have been widely used in rehabilitation robotics to ensure that the device can respond to changes in user movement. However, these methods often do not adapt well to rapid variations in interaction conditions.

2. **Machine Learning and Neural Networks**: Machine learning has been explored as a viable alternative for adaptive control. Neural networks, including LSTMs, have advantages in predicting torque and adapting to complex interactions, but few studies address their specific application for real-time torque estimation in exoskeletons. This work differs by utilizing an LSTM configured to accurately estimate interaction torque and adapt to different movement profiles and gait phases.

3. Proposed Approach

The LSTM architecture proposed in this work has a sequential structure with multiple layers, each designed to capture the temporal dynamics of human movement and accurately predict interaction torque. The network was configured with three LSTM layers of 128, 64, and 32 neurons, respectively, with dropout layers applied to prevent overfitting. The loss function used is the Huber loss, which has proven effective in modeling systems with outliers.

$$\tau_i = K_a(\theta_h - \theta_r) + B_a(\dot{\theta}_h - \dot{\theta}_r), \tag{1}$$

where K_a and B_a represent the stiffness and damping coefficients, θ_h and $\dot{\theta}_h$ are the human position and velocity, while θ_r and $\dot{\theta}_r$ are the corresponding values for the robot. The proposed architecture adapts better to rapid variations compared to traditional methods, due to the LSTM's ability to handle temporal dependencies.

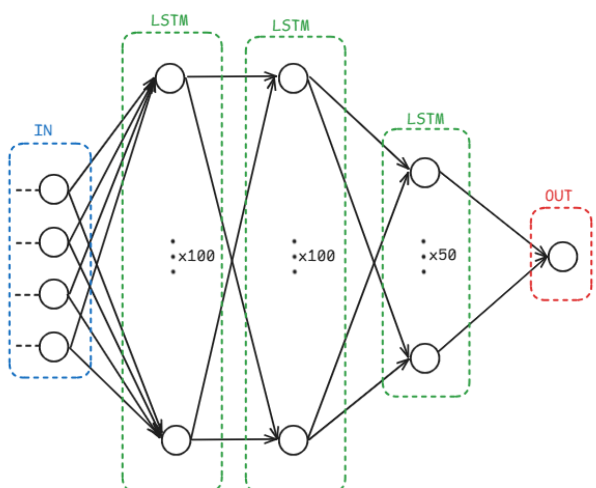

Figure 1: Pipeline diagram showing the flow of data from interaction torque to the desired torque prediction R_T.

4. Mathematical Development of Metrics

In this section, we present the mathematical development of the metrics used to evaluate the performance of the torque estimation model.

4.1. Mean Absolute Error (MAE)

Mean Absolute Error (MAE) is a metric that quantifies the average of the absolute errors between predicted values and actual values. It is defined by the following equation:

$$MAE = \frac{1}{n} \sum_{i=1}^{n} |y_i - \hat{y}_i|, \tag{2}$$

where: - y_i is the actual value, - $\hat{y}_i$ is the predicted value by the model, - n is the total number of observations.

MAE provides a direct measure of model accuracy since it proportionally penalizes each error, without amplifying the squared errors.

4.2. Mean Squared Error (MSE)

Mean Squared Error (MSE) measures the average of squared errors, being more sensitive to outliers due to squaring. Its formula is given by:

$$MSE = \frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2. \tag{3}$$

The terms are the same as those defined previously. MSE is frequently used in machine learning optimization problems because its derivatives are easy to calculate, facilitating minimization.

4.3. Coefficient of Determination (R^2)

The coefficient of determination (R^2) indicates the proportion of variability in the data that is explained by the model. It is calculated as:

$$R^2 = 1 - \frac{SS_{res}}{SS_{tot}}, \tag{4}$$

where: - $SS_{res} = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2$ is the sum of squared residuals, - $SS_{tot} = \sum_{i=1}^{n} (y_i - \bar{y})^2$ is the total sum of squares, where $\bar{y}$ is the mean of the actual values.

The value of R^2 ranges from 0 to 1, where values closer to 1 indicate a better fit of the model to the data.

5. Experiments

5.1. Experimental Setup

To validate the approach, an experimental setup was used consisting of torque sensors coupled to the exoskeleton, controlled by a Raspberry Pi 4. The collected data were processed to feed the LSTM network, with evaluation metrics including Mean Absolute Error (MAE), Mean Squared Error (MSE), and coefficient of determination R^2. The network's hyperparameters include an initial learning rate of 0.001 and a batch size of 32, adjusted based on preliminary performance analysis.

5.2. Results and Discussion

The experimental results showed that the LSTM model was able to predict the interaction torque with a significant reduction in MAE and MSE compared to traditional models. Figure 2 illustrates the comparison between the torque estimated by the model and the actual measured torque. The LSTM network's accuracy in capturing the dynamic behavior was evident, highlighting its effectiveness for application in rehabilitation robotics.

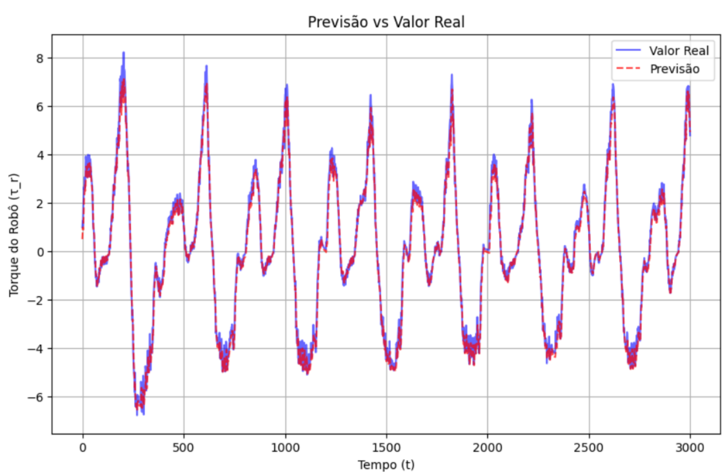

Figure 2: Comparison between estimated torque and actual torque.

6. Conclusion

This work presented an LSTM-based approach for torque estimation in rehabilitation exoskeletons, aiming to improve human-robot interaction through more precise adaptive control. The methodology demonstrated promising results, with improvements in performance and transparency of the device. As future work, it is suggested to increase the dataset and explore more complex models that can adapt to other movement profiles and individual user needs.

Appendix A. Colab Notebook Code

The complete code can be accessed at:
`github.com/EricRibeiroAlves/Interacao-Humano-Robo`.

References

[1] C. J. T. W. de Oliveira, M. H. T. S. M. de Oliveira, "Comparative analysis of assistive robotic systems for upper limbs in rehabilitation," *Journal of Robotic Systems*, vol. 38, no. 8, pp. 745-759, 2021.

[2] S. W. Chen, "Deep Learning Approaches for Human-Robot Interaction," *IEEE Transactions on Human-Machine Systems*, vol. 50, no. 1, pp. 120-130, 2020.