Hand Gesture Recognition using Liquid Neural Networks

Tapan Chiknis (211EC111)
Department of Electronics and Communication Engineering
National Institute of Technology Karnataka
Surathkal, India
chiknistapanavinash.211ec111@nitk.edu.in

K V Srinanda (211EC117)
Department of Electronics and Communication Engineering
National Institute of Technology Karnataka
Surathkal, India
kalikivenkatasrinanda.211ec117@nitk.edu.in

Suhas R P (211EC153)
Department of Electronics and Communication Engineering
National Institute of Technology Karnataka
Surathkal, India
suhasrp.211ec153@nitk.edu.in

Anirudh Bhat Nekkare (211EC268)
Department of Electronics and Communication Engineering
National Institute of Technology Karnataka
Surathkal, India
anirudh.211ee213@nitk.edu.in

***Abstract*—In the realm of real-time data processing, static neural networks often falter, particularly when tasked with handling dynamic inputs such as video or audio streams. Liquid neural networks (LNNs) have emerged as a promising solution to this challenge. LNNs offer stable and bounded behavior, enhanced expressivity within neural ordinary differential equations, and superior performance on time-series prediction tasks. In this project, we explore the effectiveness of Liquid Neural Networks (LNNs) compared to traditional Long Short-Term Memory (LSTM) models and Continuous-Time Recurrent Neural Networks (CT-RNN) in classifying hand gestures captured from sensors into five distinct classes. Through a comparative study, we demonstrate that the LNN model outperforms both LSTM and CT-RNN models in this real-time data classification task. Our findings underscore the efficacy of LNNs in handling dynamic inputs and highlight their potential for applications requiring rapid and accurate classification of sensory data.**

***Index Terms*—Long Short-Term Memory (LSTM), Liquid Neural Network (LNN), Continuous-Time Recurrent Neural Networks (CT-RNN), LTC (Liquid Time-Constant)**

I. Introduction

A neural network is a type of artificial intelligence model inspired by the structure and function of the human brain. It consists of interconnected nodes, or neurons, organized into layers. Each neuron receives input, processes it through an activation function, and produces an output. Neural networks are trained using large datasets to learn patterns and relationships within the data, enabling them to make predictions or classifications on new, unseen data. Neural networks have evolved since their inception in the 1980s, with the back-propagation algorithm being a notable learning method [1]. They are inspired by the human brain and have become the dominant branch of machine learning, with the multi-layer perceptron (MLP) model being popular [2]. Neural networks are important data mining tools used for classification and clustering, learning from examples and discovering patterns in data [3]. They are widely used in quantum chemistry in the form of combination of their different types [4]. Neural networks have gained popularity recently due to technological advancements and computational power, outperforming other machine learning algorithms [5]. CT-RNN, or Continuous-Time Recurrent Neural Network, models dynamics in continuous time, unlike traditional discrete-time RNNs. It formulates recurrent connections as continuous-time differential equations, enabling accurate modeling of systems with continuous dynamics. CT-RNNs are adept at handling irregularly sampled data and capturing long-term dependencies. They find applications in time-series prediction, control systems, and modeling dynamical systems.

A Liquid Neural Network (LNN) is a type of artificial neural network designed to handle dynamic inputs, such as continuous streams of data like video or audio. These networks are designed to adapt to changing conditions and can distill relevant information from visual inputs, allowing them to generalize their navigation skills to new environments. Liquid networks have been shown to outperform other deep agents in terms of decision-making robustness, both in their differential equation and closed-form representations [7]. Liquid Neural Networks (LNNs) are a new type of neural network specifically made for handling time-based data. Unlike regular networks that follow a straight path, LNNs have a flexible ”liquid” layer that helps them understand and use the patterns in sequences of data more effectively [8]. It draws inspiration from the behavior of liquids, where information flows smoothly and continuously. LNNs consist of interconnected nodes, similar to neurons in the human brain, but they use a different approach to processing data. However, what sets LNNs apart is their ability to process data over time using neural ordinary differential equations (ODEs). This means they can handle continuous streams of data and adapt to changes in real-time, making them particularly well-suited for tasks involving dynamic inputs. One of the key advantages of LNNs lies in their stability and bounded behavior, which ensures consistent performance even when faced with rapidly changing input data. LNNs are utilized in pattern recognition tasks, such as gesture recognition, anomaly detection, and event recognition within video streams or sensor data. They possess the capability to recognize complex patterns within dynamic data [8]. The rest of the report is divided into 2 sections, Section II Implementation where the dataset used is explained along with the data pre-processing, model architecture, and training setup, Section III Results and Discussions explains the results obtained by implementing the models.

II. Implementation

A. Dataset

The dataset comprises seven recordings capturing individuals performing a series of hand movements. Each time-step in the dataset is characterized by 32 data points collected from a motion detection sensor. At each time step, the output indicates one of five potential hand gestures: rest position, preparation, stroke, hold, or retraction. The goal is to develop a classifier capable of accurately identifying hand gestures based on the motion data.

B. Data pre-processing

Before training the model, other tasks such as data loading and preprocessing tasks, including reading sensor data from CSV files, segmenting the input data into sequences of a specified length, and dividing the data into training, validation, and test sets were handled. Each of the seven recordings was segmented into overlapping sub-sequences, each containing exactly 32 time-steps. Following segmentation, the subsequences were randomly split into non-overlapping sets for training (75%), validation (10%), and testing (15%). Input features underwent normalization to achieve zero mean and unit standard deviation. Categorical classification accuracy was employed as the performance metric for evaluating the model.

C. Model architecture and training setup

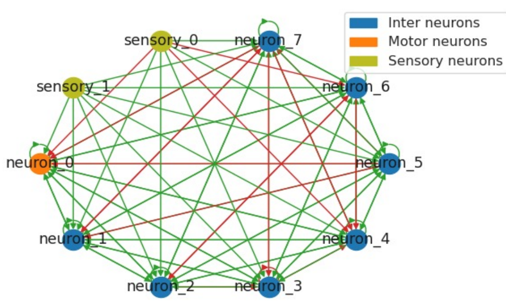

Fig. 1. Fully Connected Liquid Neural Network of 8 Neurons

The network of interconnected neurons is shown in Fig.1. It shows the fully connected liquid neural network of 8 neurons. The sensory neurons transmit signals from the body to the central nervous system. Motor neurons transmit signals from the central nervous system to the muscles. Interneurons connect sensory neurons to motor neurons, and also connect to other interneurons. In this project, we have compared the performance of three recurrent neural network (RNN) architectures: Long Short-Term Memory (LSTM), Liquid Time-Constant Networks (LTC), and Continuous-Time Recurrent Neural Networks (CT-RNN). These architectures were evaluated for their effectiveness in classifying hand gestures based on sensor data. The code utilizes TensorFlow, a widely-used deep learning framework, for its ease of use, efficient computation, and flexibility. The model architecture includes an input layer, followed by either LSTM , LTC or CT-RNN layers, and an output layer with a softmax activation function for classification. The Adam optimizer is employed to minimize the sparse softmax cross-entropy loss during training. Regularization techniques like L1 or L2 regularization are not explicitly used, but certain RNN architectures may include built-in regularization mechanisms, such as dropout, to prevent overfitting. The training process involves iterating over batches of training data, computing the loss, and updating the model parameters using backpropagation. The accuracy metric is used to evaluate the models' performance on validation and test datasets.

Our project used specific hyperparameters to train our model effectively. The model was trained in batches of 16 samples at a time to make the learning process efficient. The learning rate, which controlled how much the model learned from each batch, was set differently depending on the type of model we were using. We chose to have 32 hidden units in each layer of our model to balance complexity and efficiency. A special optimization method called Adam was used, with carefully chosen settings to update the model's weights effectively. The training lasted for 200 epochs, giving the model plenty of chances to learn from the data. For models that dealt with sequences of data, like LSTMs, we trained them to remember patterns across 32 steps. Parameters were also set for solving certain mathematical equations our model used. We regularly checked how well our model was doing on a separate set of data to make sure it was learning correctly. These settings were crucial for getting the best results in our task of classifying hand gestures.

III. Results and Discussions

TABLE I
Results obtained from implementation using 32 units in hidden layer

Model	Implemented Accuracy
LSTM	59.44%
CT-RNN	57.71%
LTC	68.3%

The comparison of the accuracy of three different models when utilizing 32 units in the hidden layer of the models is shown in Table I. The models compared were LSTM, CT-RNN and LTC. LTC [6] had achieved the highest accuracy (68.30%) among the implemented models. LSTM and CTRNN models comparatively perform worse than the LTC model.

ACKNOWLEDGMENT

We gratefully acknowledge Dr. A V Narasimhadhan for providing us with the opportunity to work on this project, thereby enriching our understanding and practical skills. His invaluable insights and guidance have significantly influenced the quality of our work.

REFERENCES

[1] Artificial Neural Networks. Advances in information quality and management, (2024). doi: 10.4018/978-1-6684-7366-5.ch037
[2] Neural Networks, (2023). doi: 10.1017/9781107588493.007
[3] Understanding Neural Networks, (2023). doi: 10.1002/9781119865216.ch4
[4] Pavlo, O., Dral., Nicolas, Dupin. (2023). Neural networks. doi: 10.1016/b978-0-323-90049-2.00011-1
[5] Neural Networks, (2023). doi: 10.1007/978-1-4842-8754-5_10
[6] Hasani, R.; Amini, A.; Lechner, M.; Grosu, R.; and Rus, D. (2020). Liquid Time-constant Networks. doi: 10.48550/arXiv.2006.04439
[7] Makram, Chahine., Ramin, M., Hasani., Aaron, Parker, Ray., Mathias, Lechner., Alexander, Amini., Daniela, Rus. Robust flight navigation out of distribution with liquid neural networks. Science robotics, (2023). doi: 10.1126/scirobotics.adc8892
[8] K. Kumar, A. Verma, N. Gupta and A. Yadav, "Liquid Neural Networks: A Novel Approach to Dynamic Information Processing," 2023 International Conference on Advances in Computation, Communication and Information Technology (ICAICCIT), Faridabad, India, 2023, pp. 725-730, doi: 10.1109/ICAICCIT60255.2023.10466162.