Dynamic Criminal Network Link Forecasting via Deep Reinforcement Learning

Srikant Tayade[1], Satish Kale[2], Soham Patil[3], Atharva Shinde[4] , Ishan Shah[5] , Omkar Pawar[6]

[1] *Scientist E, C-DAC, Pune, India*
[2] *Professor, AI&DS, AISSMS IOIT, Pune, India*
[3] *Student, AI&DS, AISSMS IOIT, Pune, India*
[4] *Student, AI&DS, AISSMS IOIT, Pune, India*
[5] *Student, AI&DS, AISSMS IOIT, Pune, India*
[6] *Student, AI&DS, AISSMS IOIT, Pune, India*

ABSTRACT

Criminal network analysis (CNA) presents unique issues due to the clandestine nature of such networks, resulting in incomplete datasets with missing nodes (actors) and linkages (relationships). Traditional link prediction methods, often based on social network analysis (SNA) and supervised machine learning (ML) approaches, struggle to attain high accuracy in this domain due to their reliance on big datasets and static snapshots of network data. To address these limitations, we use deep reinforcement learning (DRL) to predict links in criminal networks. Our time-based DRL model (TDRL) learns from an evolving dataset. The TDRL model, which generates synthetic data through self-play or self-simulation, requires less data and adapts to dynamic network changes, outperforming traditional supervised methods like gradient boosting. Regarding forecasting accuracy. Our model uses natural language processing-inspired embedding approaches to improve computing efficiency during training..

Keyword *Deep Reinforcement Learning, Criminal Network Analysis , Neural Network , Deep Q-Network*

---

1. INTRODUCTION

A vital technique for comprehending and forecasting the behavior of covert networks—where players and their connections are frequently hidden—is criminal network analysis, or CNA. Traditional link prediction techniques, which are mainly based on social network analysis (SNA) and conventional machine learning (ML) methods, struggle to accurately predict hidden or potential interactions within a criminal network because of the incomplete and inconsistent nature of the data within these networks. Due to data scarcity and privacy concerns, these models often demand huge datasets for effective training, which is impractical in this sector .

In order to overcome these constraints, this study investigates a unique method for link prediction in criminal networks by utilizing deep reinforcement learning (DRL). By producing self-play or self simulated data, DRL has demonstrated promise in allowing link prediction models to train efficiently on fewer datasets while adjusting to the temporal dynamics of the network. The main objective is to create a time-dependent link prediction model (TDRL) based on DRL that can be updated continuously as the structure of the criminal network changes. This method, in contrast to conventional supervised machine learning models, can learn from sparse data and offer high forecast accuracy for dynamic and ever evolving criminal networks.

Our model seeks to outperform conventional approaches in terms of predictive accuracy and adaptability by combining strategies like graph embedding, which converts network information into continuous vectors, and NLP-inspired approaches like DeepWalk and Node2Vec, which captur neighborhood similarity. Furthermore, strong privacy safeguards are put in place to safeguard private data in the dataset, which makes this strategy workable for practical uses. This study shows that DRL can perform better than traditional link prediction techniques, pointing to a possible avenue for further investigation in criminal network analysis and other fields where networks are dynamic and data is scarce.

1.1 LITERATURE SURVEY

1: Criminal Network Analysis (CNA) and Challenges. illicit networks are clandestine and highly dynamic, with incomplete or unreliable data due to the hidden nature of illicit activity. Traditional SNA measures including as centrality, betweenness, and clustering coefficients have long been employed to investigate criminal networks. However, these measurements fail to represent the dynamic character of criminal networks, in which relationships between persons (links) may develop, disappear, or change over time. Researchers like Sparrow (1991) and Xu & Chen (2005) have highlighted the particular obstacles in researching criminal networks, such as inadequate data and the presence of hidden nodes, which complicate the prediction tasks in these networks. Link Prediction Methods** Early link prediction in networks frequently used heuristic or similarity-based methods, such as Common Neighbors, Jaccard Coefficient, and Adamic-Adar scores (Liben-Nowell & Kleinberg, 2007). 2: These strategies are based on the assumption that nodes with similar or high connectivity are more likely to form linkages. However, these approaches are largely static and do not adjust to temporal changes in the network. Reducing their effectiveness in dynamic criminal networks where connections change often. On the other hand, feature-based models** represent nodes in low-dimensional vector spaces using methods such as matrix factorization or embedding techniques (e.g., Node2Vec, DeepWalk). These techniques have been effectively used in a variety of network types and capture more intricate structure information. They still have trouble capturing temporal dynamics, though, because they rely on static snapshots, which are less appropriate for real-time applications where connections and relationships change all the time.

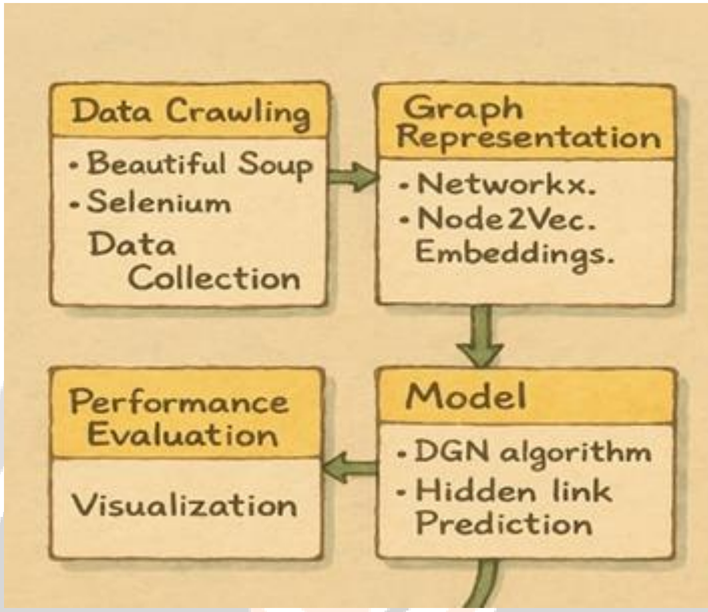
3: Preditemporal Graph Models with Dynamic Links By adding time as a crucial factor, dynamic link prediction techniques, such as temporal network analysis, aim to overcome the drawbacks of static models. To capture temporal relationships in evolving networks, methods including Time-Aware GNNs, Long Short-Term Memory (LSTM) networks, and Recurrent Neural Networks (RNNs) have been developed. A dynamic graph embedding model (DyRep), for example, was presented by Goyal et al. (2018) in order to identify temporal patterns in social networks. Even while these models work effectively, they are usually supervised and need a lot of past interaction data, which isn't often available in criminal network environments. Furthermore, techniques like GraphSAGE, GCNs, and GATs have been expanded to take temporal information into consideration, and Temporal Grapetworks (TGNNs) have recently been proposed to handle time-evolving graphs. These models perform at the cutting edge of link prediction tasks, but they are susceptible to overfitting in sparse or unbalanced datasets, which is a prevalent problem in criminal networks where certain associations are rarely seen.

4: Prediction using Deep Reinforcement Learning (DRL) New possibilities for link prediction have been made possible by recent developments in Deep Reinforcement Learning (DRL), particularly in situations with sparse data or extremely dynamic environments. Large labeled datasets are not necessary for DRL-based models to learn; instead, they can do so by self-simulation or environmental interaction. In a number of applications, including social networks, recommendation systems, and multi-agent systems, this method has demonstrated encouraging outcomes. Without requiring a large amount of pre-labeled data, DRL methods like Proximal Policy Optimization (PPO), Policy Gradient, and DQN (Deep Q-Network) enable models to learn optimal policies over time, enabling adaptive responses to network changes. Because DRL simulates expected future encounters, it has tict connections in the context of criminal networks. DRL models may effectively detect missing links based on previous interactions, as demonstrated by Peng et al.'s (2020) application of DRL for dynamic link prediction in knowledge graphs. Due to its versatility, DRL is a good fit for criminal network forecasting, where links and activity are frequently erratic and subject to rapid change.

5: Using DRL in conjunction with Graph Neural Networks (GNNs) for Dynamic Linon Combining DRL with Graph Neural Networks (GNNs) has become a potent method for capturing both spatial , temporal and structural data in dynamic networks. Node embeddings and neighborhood information are typically extracted using Graph Convolutional Networks (GCNs), Graph Attention Networks (GATs), and GraphSAGE. These can subsequently be fed into a DRL agent to learn link prediction strategies. Research has demonstrated that by learning node representations and interaction patterns over time, the combination of GNNs and DRL allows for more accurate link prediction in dynamic environments (Wang et al., 2021). Such an approach can assist criminal networks in identifying possible correlations or connections that might not be visible using only typical SNA measures.

1.2 METHODOLOGY

This section describes the thorough process used to create a deep learning-based link prediction system for criminal networks. There are multiple steps in the process, starting with data collection and preprocessing and ending with model creation, training, assessment, forecasting.



1. Work-Flow

1: Data Gathering and Preparation Network Structure: This comprises details about nodes, which are people or organizations that are a part of the criminal network, and edges, which are connections or exchanges between nodes. The link prediction task is based on the network structure. Data Collection: Compiling pertinent data is the initial stage in creating a criminal network link prediction model. The information includes a number of essential elements: Node qualities: Node qualities, which include demographics, criminal history, role in the network, and behavioral traits (e.g., frequency of interactions, geographic location), are significant in addition to the network structure. These characteristics offer more detailed information about the network's actors. Temporal Information: Relationships within criminal networks frequently change over time, making them dynamic. Understanding the temporal component of network changes and forecasting future interactions require temporal data, such as timestamps of interactions or shifts in criminal behavior.

Data cleaning: Information gathered from criminal networks is frequently noisy and lacking. Data cleaning includes the following: • Managing Missing Values: Incomplete records may result in missing information, which is addressed by methods including imputation, elimination, or interpolation. Outlier Detection: Analyses may be distorted by anomalies or outliers in the dataset. Outliers can be found and controlled using statistical techniques or anomaly detection based on machine learning. Resolving Inconsistencies: Inconsistencies may be introduced by various sources or dataset components (e.g., varied formats for timestamps). The dataset is guaranteed to be uniform once these discrepancies are standardized and fixed. Feature engineering is the process of developing significant features that can represent the behavioral and structural traits of nodes and edges in a network. Among the key characteristics are: Node Degree: A node's number of direct connections can serve as a gauge of its significance or level of impact within the network. Clustering Coefficients: Indicates how often it is for nodes in a network to group together or form triangles. Nodes with high clustering are more likely to create close-knit groups, which may be correlated with interactions in the future.

Other Network Metrics: Depending on the network, other metrics like eigenvector centrality, closeness centrality, or page rank might also be considered to capture the importance of a node in the network's structure.

Training Set: Used to train the model, typically representing 70-80% of the total data.

Validation Set: A smaller portion used to tune model parameters and avoid overfitting.

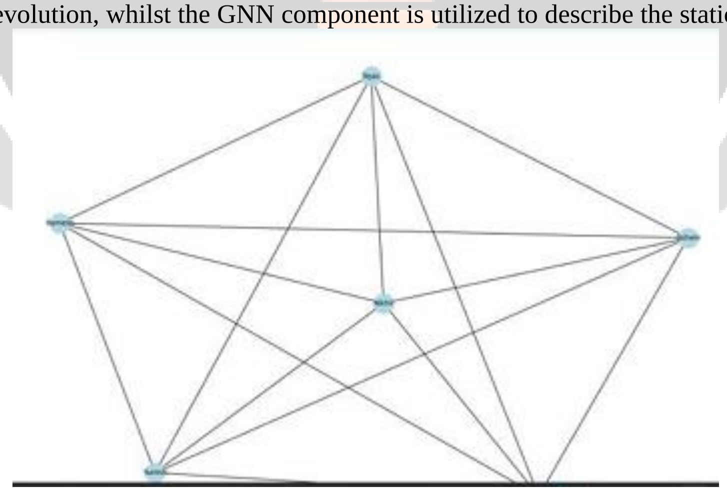
Test Set: The remaining data used to evaluate the final model's performance on unseen data.

Training Set: Used to train the model, typically representing 70-80% of the total data.

Validation Set: A smaller portion used to tune model parameters and avoid overfitting.
Test Set: The remaining data used to evaluate the final model's performance on unseen data.

2: Representation of Networks Adjacency Matrix: The adjacency matrix is one method of converting the criminal network into a mathematical representation. Each member (i, j) in the adjacency matrix, which is a square matrix, indicates whether or not there is a link connecting nodes i and j. This form, which is frequently employed in graph based deep learning techniques, aids in capturing the network's structure. Graph Embeddings: Graph embeddings are created to more compactly represent the network's high dimensional structural information. The network's nodes or subgraphs are mapped to low-dimensional vector spaces using these embeddings. Typical embedding methods include the following : Node2Vec: Using random walks, this method determines how similar nodes are to one another based on their immediate neighborhoods. By determining which nodes, in terms of network topology, are closest to one another, it creates vector representations. Deep Walk: Like Node2Vec, Deep Walk uses random walks throughout the graph before learning node embeddings using Skip-gram models, which are frequently used in NLP tasks. By converting them into continuous vectors, it effectively captures structural linkages by converting the graph structure into a format suitable for deep learning models, these embedding techniques enable the model to effectively capture intricate patterns inside the network.

3: Model for Deep Learning Graph Attention Networks (GATs): By adding attention mechanisms that give neighbors varying weights according on their significance, GATs outperform GCNs. In complicated criminal networks, where some participants may have more sway, this enables the model to concentrate more on important nodes and less on irrelevant ones. Given their ability to handle graph-structured data, graph neural networks (GNNs) are a logical choice for link prediction in criminal networks. Among the GNN types used in this work are: Graph Convolutional Networks (GCNs): GCNs use data from a node's neighbors to learn node representations. It is efficient at capturing local network architecture since each node's embedding is updated by fusing its current feature vector with the feature vectors of its neighbors. By sampling and aggregating data from a node's neighbors, the inductive model known as GraphSAGE (Graph Sample and Aggregation) develops node representations. This makes it more capable of managing big networks and invisible nodes. Recurrent neural networks, or RNNs, are utilized to represent the sequential or time-dependent character of temporal criminal networks. In order to forecast future interactions, it is essential to comprehend how the relationships between nodes change over time. The RNN model offers a dynamic perspective of the criminal network by updating the network's state. Hybrid Models: These models combine the advantages of GNNs and RNNs to capture the network's temporal and structural features. The RNN component is in charge of modeling the network's temporal evolution, whilst the GNN component is utilized to describe the static graph structure.



2. Criminal Links

Here ,Nodes are names of linked criminals and Edges shows relationship between them .

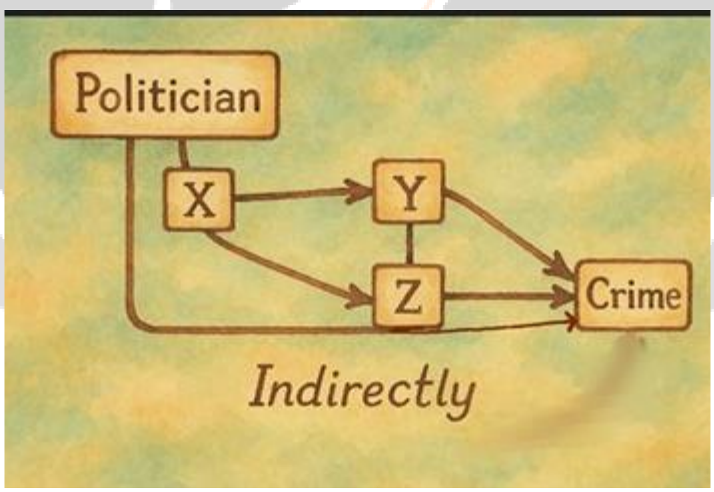
4 : Training and Assessing Models Accuracy: Indicates the percentage of accurate forecasts.Precision: Calculates the percentage of all positive forecasts that are actually positive. Recall: Calculates the percentage of actual positive occurrences that were accurate positive predictions. When there is an imbalance between the classes, the F1-Score—a harmonic mean of precision and recall—can be helpful. AUC-ROC (Area Under the Curve-Receiver Operating Characteristic): This performance statistic, which is independent of class distribution, assesses the Loss Function: A

binary cross-entropy loss function is used to train the link prediction model. Because it calculates the difference between a link's actual presence or absence in the network and its expected probability, this function is appropriate for binary problems. Optimization: To minimize the loss function, optimization procedures are applied. Some popular optimizers are the Adam Optimizer, an adaptive optimization technique that offers quick convergence by combining the advantages of momentum and RMSprop. Another adaptive learning rate technique that aids in training process stabilization, particularly in non convex optimization situations, is RMSprop. Evaluation Metrics: A number of common metrics are used to assess the model's performance: Accuracy: Indicates the percentage of accurate forecasts. Precision: Calculates the percentage of all positive forecasts that are actually positive. Recall: Calculates the percentage of actual positive occurrences that were accurate positive predictions. When there is an imbalance between the classes, the F1-Score—a harmonic mean of precision and recall—can be helpful. AUC-ROC (Area Under the Curve-Receiver Operating Characteristic): This performance statistic, which is independent of class distribution, assesses the model's capacity to distinguish between positive and negative data.

5: © April 2025 | IJIRT | Volume 11 Issue 11 | ISSN: 2349-6002 Predicting Links: Pairs of nodes, which stand in for possible network links, are fed into the model. These node pairs may be chosen at random or in accordance with suspicious activity or historical data. Output: The likelihood that a link will eventually exist between the nodes is predicted by the model. Based on the two nodes' characteristics and their connections inside the network, as determined by the deep learning model, this probability represents the possibility of interaction between them. In conclusion, the approach presented here builds a strong link prediction system for criminal networks by integrating cutting-edge methods in deep reinforcement learning, temporal modeling, and graph neural networks. This method provides a potential way to forecast hidden linkages and increase the accuracy of criminal network analysis by addressing the problems of missing data, dynamic changes, and the complexity of criminal network structures.

2. EXAMPLE

"X" is a criminal in Tihar jail with "Y " in same jail.
"Y" was previously with "A" in Central jail.
"B" is "A"'s brother and he works for a Politician.



3. Example Link

1. Politician wants to commit a murder without letting people know he's involved in that, so he tells "B" to find a person who can commit crime and if he gets caught politican won't be caught.
2. "B" finds a older cell-mate "A" to perform crime but he's in jail under a case so he cannot perform crime , so he suggests his brother "Y" .
3. "Y" is also in jail so he suggests "X" who is going to get released soon and its a deep network so politician won't be found out.
4. "X" commits crime lead by the Politican.

3. EQUATION

Mathematical formula (DQN):

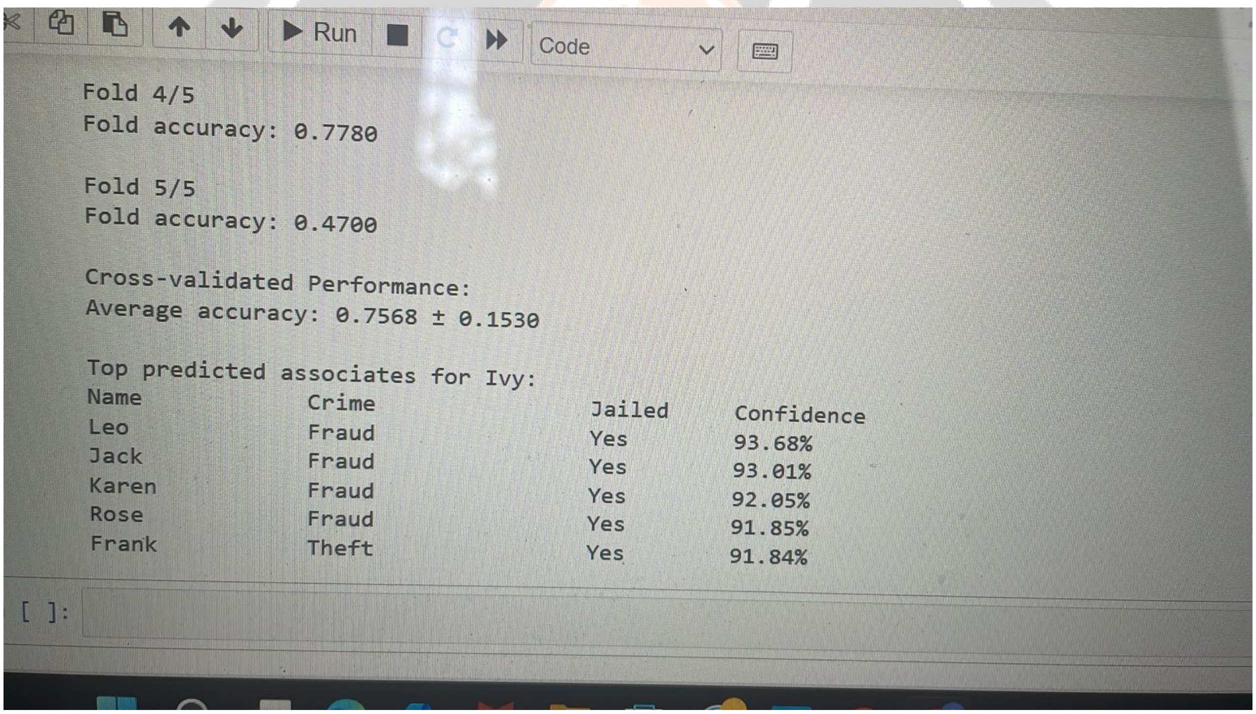
$$Q(s,a) = \overbrace{Q(s,a)}^{\text{Old value}} + \underbrace{\alpha}_{\text{learning rate}}(\overbrace{r}^{\text{reward}} + \underbrace{\gamma}_{\text{discount}} \overbrace{maxQ(s'a')}^{\text{optimal future value}} - \overbrace{Q(s,a)}^{\text{Old value}})$$

4. Q – Learning Formula

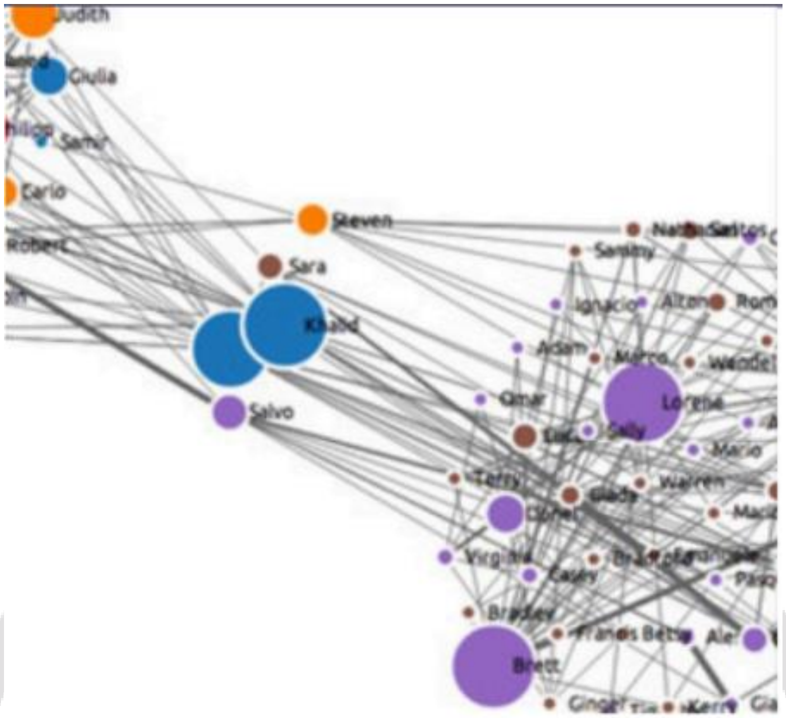
4. CONCLUSIONS

We conclude that by learning from the dynamic character of criminal networks, Deep Reinforcement Learning (DRL) for link prediction offers a potent tool to predict future connections. DRL continuously adjusts to changing behavioral patterns, increasing the precision of its predictions. By combining DRL and Transfer Learning, the model can benefit from expertise from related fields, improving its capacity to identify new patterns and dangers. By proactively preventing crimes before they worsen, this strategy aids law enforcement in anticipating criminal activity .

5. OUTPUT



5. Accuracy, Confidence



6. 3D representation

6. REFERENCES

[1] Zhang, Yi, Xiaodong Zhang, Zhufeng Lu, Zhiming Jiang, and Teng Zhang. "A Novel Wrist Joint Torque Prediction Method Based on EMG and LSTM." International Conference on Cyber Technology in Automation, Control, and Intelligent Systems, pp. 242- 245, 2020.

[2] Matsumura, Yuji, Minoru Fukumi, and Norio Akamatsu. "Wrist motion pattern recognition system by EMG signals." International Conference on Knowledge-Based and Intelligent Information and Engineering Systems, pp. 611-617, 2005.

[3] Khokhar, Zeeshan O., Zhen G. Xiao, and Carlo Menon. "Surface EMG pattern recognition for real- time control of a wrist exoskeleton." Biomedical engineering online 9, no. 1 pp.1-17, 2010. Parajuli, N.; Sreenivasan, N.; Bifulco, P.; Cesarelli, M.; Savino, S.; Niola, V.; Esposito, D.; Hamilton, T.J.; Naik, G.R.; Gunawardana, U.; Gargiulo, G.D. “Real- Time EMG Based Pattern Recognition Control for Hand Prostheses”: A Review on Existing Methods, Challenges and Future Implementation. Sensors pp.4596-4598, 2019.

[4] Parajuli, Nawadita, Neethu Sreenivasan, Paolo Bifulco, Mario Cesarelli, Sergio Savino, Vincenzo Niola, Daniele Esposito et al. "Real-time EMG based pattern recognition control for hand prostheses: A review on existing methods, challenges and future implementation." Sensors 19, no. 20, pp.4596-4600, 2019.

[5] Selvan, Mercy Paul, Rishi Raj, R. Gowtham Sai, S. Jancy, and Viji Amutha Mary. "Prosthetic hand using EMG."Journal of Physics: Conference Series, vol. 1770, no. 1, pp.12-18., 2021.

[6] Qichuan, Ding, Zhao Xingang, and Han Jianda. "A hybrid EMG model for the estimation of multijoint movement in activities of daily living." International Conference on Multisensor Fusion and Information Integration for Intelligent Systems, pp. 1-6, 2014.

[7] Liu, Jie, Yupeng Ren, Dali Xu, Sang Hoon Kang, and Li-Qun Zhang. "EMG-based real-time linear nonlinear cascade regression decoding of shoulder, elbow, and wrist movements in able-bodied persons and stroke survivors." IEEE Transactions on Biomedical Engineering 67, no. 5 pp.1272-1281, 2019.

[8] Li, Guanglin, Oluwarotimi Williams Samuel, Chuang Lin, Mojisola Grace Asogbon, Peng Fang, and Pau Oluwagbengba Idowu. "Realizing efficient EMG- based prosthetic control strategy." Neural Interface: Frontiers and Applications, pp.149-166, 2019.

[9] A. David Orjuela-Cañón, A. F. Ruíz-Olaya and L. Forero, "Deep neural network for EMG signal classification of wrist position: Preliminary results," 2017 IEEE Latin American Conference on Computational Intelligence, pp. 1-5, 2017.

[10] Orjuela-Cañón, Alvaro David, Andrés F. Ruíz-Olaya, and Leonardo Forero. "Deep neural network for EMG signal classification of wrist position: Preliminary results."Latin American Conference Computational Intelligence, pp. 1-5, 2017.

[11] on Ziai, Amirreza, and Carlo Menon. "A linear regression model for estimation of isometric wrist joint torques with varying arm configurations using EMG signals." International Conference on Robotics and Biomimetics, pp. 1230-1235, 2011 .

[12] Kim, Sehyeon, Dae Youp Shin, Taekyung Kim, Sangsook Lee,, Jung Keun Hyun, and Sung-Min Park. "Enhanced Recognition of Amputated Wrist and Hand Movements by Deep Learning Method Using Multimodal Fusion of Electromyography and Electroencephalography." Sensors 22, no. 2 pp.680- 685, 2022 .

[13] Kirsch, R. F., and J. G. Hincapie. "Feasibility of EMGbased control of arm movements via FNS." 25th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, vol. 2, pp. 1471-1474, 2003.

[14] Pan, Lizhi, Dustin L. Crouch, and He Huang. "Comparing EMG-based human-machine interfaces for estimating continuous, coordinated movements." IEEE transactions on neural systems and rehabilitation engineering 27, no. 10 pp. 2145-2154, 2019 .

[15] Fukuda, Osamu, Toshio Tsuji, Makoto Kaneko, and Akira Otsuka. "A human-assisting manipulator teleoperated by EMG signals and arm motions." IEEE transactions on robotics and automation 19, no. 2 pp.210-222, 2003.

[16] Kurisu, Naoyuki, Nobutaka Tsujiuchi, and Takayuki Koizumi. "Prosthetic hand control using motion discrimination from EMG signals."Annual International Conference of the IEEE Engineering in Medicine and Biology Society, pp. 6922-6925, 2009.

[17] Lima, Aron A. Martins, Rafael M. Araujo, Fábio A. Guidotti dos Santos, Victor H. Yoshizumi, Fábio KH de Barros, Danilo H. Spatti, Luisa HB Liboni, and Maria E. Dajer. "Classification of hand movements from emg signals using optimized mlp." International Joint Conference on Neural Networks (IJCNN), pp. 1- 7, 2018.

[18] Kilic, Ergin. "EMG based neural network and admittance control of an active wrist orthosis." Journal of Mechanical Science and Technology 31, no. 12 pp. 6093-6106, 2017.

[19] Kawano, Takuhei, and Koichi Koganezawa. "A method of discriminating fingers and wrist action from surface EMG signals for controlling robotic or prosthetic forearm hand." In 2016 IEEE International Conference on Advanced Intelligent Mechatronics, pp. 13-18, 2016.