

Article

Enhancing Communication in CPS Using Graph-Based Reply Relationship Identification in Multi-Party Conversations

Bingwei Zhu ^{1,2}, Jinzhu Yang ³, Lirong Qiu ^{1,2}, Weichun Sun ⁴ and Bin Hou ^{3,*}

¹ School of Computer (National Demonstrative School of Software), Beijing University of Posts and Telecommunications, Beijing 100876, China; zhubingwei@bupt.edu.cn (B.Z.); qiulirong@bupt.edu.cn (L.Q.)

² Key Laboratory of Trustworthy Distributed Computing and Service (BUPT), Ministry of Education, Beijing 100876, China

³ National Internet Emergency Center, Beijing 100029, China; yangjinzhu@cert.org.cn

⁴ School of Intelligence Science and Technology, University of Science and Technology Beijing, Beijing 100083, China; u202343410@xs.ustb.edu.cn

* Correspondence: houbin@cert.org.cn; Tel.: +86-150-1029-8611

Abstract: To enhance communication and collaborative work efficiency in cyber–physical systems (CPSs) within the Industry 4.0 environment, this study investigates a graph-based machine learning approach aimed at optimizing information interaction during multi-party conversations. Devices within CPSs must efficiently exchange information in real time to synchronize operations and responses. This research treats these interactions as intricate graph structures and uses graph learning techniques to accurately identify communication links and dependencies among devices. This improvement leads to more accurate decision-making and smoother operations. Our methodology involves a real-time analysis of structural patterns and node attributes within conversations, improving information flow and comprehension. The empirical findings demonstrate that this approach significantly enhances production efficiency, system adaptability, and minimizes delays attributed to communication misunderstandings. Our method can effectively identify the communication relationships between devices, significantly improving the efficiency and accuracy of information transmission. This improved communication capability leads to an enhanced production efficiency of the entire system.

Keywords: graph-based machine learning; industrial automation; real-time communication; system synchronization



Citation: Zhu, B.; Yang, J.; Qiu, L.; Sun, W.; Hou, B. Enhancing Communication in CPS Using Graph-Based Reply Relationship Identification in Multi-Party Conversations. *Electronics* **2024**, *13*, 2302. <https://doi.org/10.3390/electronics13122302>

Academic Editor: Christos J. Bouras

Received: 30 April 2024

Revised: 27 May 2024

Accepted: 31 May 2024

Published: 12 June 2024



Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Establishing clear and reliable communication channels is paramount within Industry 4.0's cyber–physical systems (CPSs). While CPSs are widely used in Industry 4.0, there are still gaps in current research on their communication efficiency. Specifically, the use of graph learning technology to optimize CPS communication needs further exploration. This need is even more critical in environments characterized by prevalent multi-party conversations (MPCs). These interactions are vital for day-to-day operations, network security, and fostering a positive communication ecosystem. Identifying reply relationships within these MPCs is crucial as they outline interaction patterns and information flow among devices. This analysis can be compared to user interactions on digital platforms [1], highlighting its importance in mitigating risks such as information manipulation and identifying potential threats or operational anomalies. Ultimately, this capability is key to safeguarding the integrity and privacy of the systems involved.

In recent years, the application of graph learning technology in CPSs has attracted widespread attention. Presekal et al. [2], adopted a hybrid deep learning model of graph convolutional long short-term memory (GC-LSTM) and a deep convolutional network for anomaly detection based on time series classification. Wu et al. [3] proposed a graph

attention unit network (PGRGAT) for unsupervised anomaly detection. The application of graph convolutional networks in the industrial Internet of Things is explored to enable efficient data transmission and fault detection between devices.

Advanced graph learning technologies, such as graph convolutional networks (GCNs) [4], graph attention networks (GATs) [5], and graph autoencoders [6], can effectively capture patterns and relationships in complex networks by leveraging adjacency relationships and node features in graph structures. Applying these technologies to cyber-physical systems (CPSs) can enhance both communication efficiency and data processing capabilities. In this study, “conversations between CPSs” refers to the data exchange and communication processes between various CPS components. By utilizing graph learning technology, we can better understand and optimize these communication processes, thereby improving the overall efficiency of the system.

In situations vulnerable to security threats, such as cyber-attacks or technical failures, the complexity of participant interactions and the dynamic nature of topics in multi-party conversations (MPCs) present significant challenges. These intricate interactions often hinder the identification of response relationships, making it difficult for conventional systems to facilitate prompt interventions [7,8]. Additionally, traditional approaches [9–12] generally fail to adequately consider the critical role of dialogue structure in examining such interactions. This oversight can undermine both operational integrity and security. This paper presents a novel approach that involves transforming MPCs into graph structures and utilizing advanced graph learning techniques to effectively discern and leverage structural data.

In cyber-physical systems (CPSs), conversations are represented as comprehensive graph structures. Each message or signal is treated as a node, and links are established between nodes that share reply relationships. This mapping allows for the precise identification of reply relationships through a link prediction task. By using link prediction technology, we can accurately identify potential communication relationships between devices in the CPS communication network. This optimization of information transfer paths reduces communication delays and improves the overall efficiency of the system. This is especially important in an Industry 4.0 environment, where real-time and efficient information transfer is essential for the normal operation of smart manufacturing systems. Additionally, reinforcement learning is integrated to dynamically determine the optimal subgraph size, enabling adaptation to the complexity of dialogue structures.

In the context of Industry 4.0, smart manufacturing systems rely heavily on efficient communication among various cyber-physical system (CPS) components such as robots, sensors, and control systems. These components must synchronize their operations in real time to maintain the optimal production flow, minimizing delays and errors. Consider an automated production line in a smart manufacturing workshop. This production line comprises multiple robotic workstations, each responsible for different production steps. These robots communicate through sensors and control systems, exchanging information about production status, material positions, and processing progress. Traditional communication methods currently in use face issues such as high latency, information loss, and miscommunication, which adversely affect production efficiency. Our proposed graph-based machine learning model addresses these communication challenges by treating the interactions between CPS components as a graph. In this graph, each robot or sensor is represented as a node, and the communication links between them are represented as edges. By applying advanced graph learning techniques, our model can accurately identify and predict the optimal communication pathways, enhancing the reliability and efficiency of information exchange.

Experiments on multi-party conversation datasets and link prediction datasets show that our model surpasses other models across various evaluation metrics. The main contributions of this study are as follows:

1. We introduce a secure framework for analyzing multi-party conversations (MPCs) in cyber-physical systems (CPSs). This is achieved by converting the task of iden-

tifying reply relationships into a link prediction challenge using advanced graph learning techniques.

2. We present a novel subgraph extraction method that utilizes reinforcement learning to adapt to the dialogue's complexity, enhancing model training efficiency and reducing resource usage.
3. Our extensive experimental validation demonstrates the superior performance of our proposed method in identifying response relationships in MPCs and its applicability to other link prediction tasks.

The organization structure of this paper is as follows: Section 2 introduces related work; Section 3 introduces the component modules and overall structure of Policy-SEAL; Section 4 introduces the algorithm implementation details of Policy-SEAL; Section 5 explains and analyzes the experimental results; and Section 6 finally summarizes the full text.

2. Related Work

This article focuses on methods for solving message reply relationship judgments in multi-party conversations, covering the fields of multi-party conversations and link prediction. Therefore, this chapter will further explore related research on MPCs and link prediction tasks.

2.1. Multi-Party Conversations

In the field of multi-party conversation (MPC), dialogue disentanglement and response generation are two crucial tasks, along with the evaluation of reply relationships. The initial exploration of reply relationships involved creating an MPC corpus to investigate recipient and response selection tasks (Ouchi and Tsuboi [9]). This endeavor primarily focused on parent nodes to pinpoint the "reply-to" relationship between sentences by employing a method that integrates conversational history through a two-layer transformer structure, reinforced by a masking mechanism (Zhu et al. [10]). In the field of dialogue disentanglement, researchers have used co-training methods to achieve unsupervised disentanglement. This method uses a message pair classifier to identify the local relationship between two messages and a session classifier to capture contextual awareness, classifying messages into dialogues (Liu et al. [11]). The structure of the dialogue is further highlighted by a newly developed model for dialogue disentanglement (Ma et al. [12]). Furthermore, integrating interaction features of the same speaker into the session context representation during the second phase of pre-training, using a Structured Conversational Interaction Joint Encoding (SCIJE) approach, is crucial for discerning discourse in multi-party conversations (MPCs) (Yu et al. [13]). Moving on to response generation, the HeterMPC model, introduced by Gu et al. [14], presents a pioneering approach to modeling the intricate network of interactions among utterances and participants in multi-party conversations (MPCs) using heterogeneous graphs. This model stands out by utilizing two distinct node types to concurrently represent utterance semantics and participant dynamics. Notably, HeterMPC leverages six types of meta-relations to precisely map interactions within the heterogeneous graph, effectively harnessing dialogue structural knowledge for generating coherent responses. This methodology underscores the pivotal role of heterogeneous graph modeling in enhancing the accuracy of response generation and comprehension in MPC settings. Prior studies in the MPC domain have been limited by a lack of emphasis on learning structural features, focusing solely on modeling either utterance nodes or participant nodes. This oversight has led to the neglect of the rich structural information present within dialogue graphs. Consequently, these approaches are not directly applicable to the tasks under examination in this study. Although existing research has made progress in multi-party conversation analysis, current methods still struggle with the structural characteristics of complex multi-party conversations. Most methods focus only on the semantic analysis of text content, ignoring the deep structural relationships in conversations. Therefore, an innovative method that can effectively capture the structural characteristics of conversations is needed.

2.2. Link Prediction

Graph neural networks (GNNs) are commonly used for link prediction tasks, where entities and their relationships are represented in a graph structure. GNNs excel by learning and predicting from this graph structure. In CPSs, graph learning technology has been widely employed to optimize information interaction and enhance system security. El-hanashi et al. (2023) [15] reviewed the application of deep learning in the Internet of Things (IoT), highlighting the advantages of graph learning technology in managing complex network relationships. These studies provide valuable references and foundations for our methods. In our study, we leverage a graph structure to assess the relationships in MPC replies, drawing inspiration from link prediction tasks. Two prominent GNN-based link prediction methods are graph autoencoders (GAEs) [6] and SEAL [16,17]. GAEs utilize GNNs to compute node representations across the entire network and then merge these representations to predict the target link. In contrast, SEAL (Subgraph Embedding for Link Prediction) employs GNNs on subgraphs that encapsulate individual links. SEAL extracts a closed subgraph (h-hop subgraph) around each link of interest and applies dual-radius node labeling (DRNL) to assign integer labels to nodes within the subgraph. This labeled subgraph is then fed into the GNN to learn the links and estimate the probability of link presence. Although both GAEs and SEAL operate on GNN principles, SEAL consistently demonstrates superior practical performance compared to GAEs.

Link prediction is a crucial aspect of enhancing communication in CPSs by enabling the system to accurately identify and establish necessary communication links between devices. This approach leverages graph learning techniques to optimize information interaction paths, reduce communication delays, and improve data transmission accuracy. By dynamically predicting communication needs, link prediction helps in the real-time adjustment of network configurations [18,19], thus ensuring efficient and reliable operations within the CPS.

3. Optimizing CPS Communications Using Graph Learning Technology

Traditional MPC analysis methods rely on semantic analysis and linear structure processing [9–12], which fall short in handling complex MPCs due to their inability to capture the graph structure characteristics of dialogues and the deep correlations between messages. Our research aims to use graph learning technology, specifically link prediction, to optimize CPS communication in the Industry 4.0 environment. By analyzing structural patterns and node attributes in real time, link prediction dynamically adjusts network configurations, discovers and resolves communication bottlenecks promptly, and improves system performance. Applying this method reduces communication delays, improves data transmission accuracy, and enhances the system's adaptive capabilities.

This paper introduces a novel approach to address the limitations of identifying MPC reply relationships by framing it as a link prediction problem and utilizing graph learning technology. Initially, a multi-party conversation is modeled as a graph structure in which individual messages are represented as nodes and the relationships between messages as edges. This representation allows for the visualization of structural features and intricate interaction patterns within conversations. Leveraging the link prediction framework, this paper then incorporates graph learning technology, particularly graph neural networks (GNNs), to enable the learning of message node representations. The GNN is adept at capturing both local and global information embedded in the graph structure, enhancing the identification of reply relationships. Moreover, this paper employs a reinforcement learning strategy to automatically optimize the subgraph extraction approach. This strategy is crucial for improving the efficiency and accuracy of predicting reply relationships in MPCs, given the sparse and imbalanced nature of such relationships in this context.

In this article, we propose a general learning framework called Policy-SEAL, which focuses on understanding interactive relationships depicted in graphs, specifically for determining reply relationships in multi-party conversations (MPCs). The core objective of Policy-SEAL is to automatically extract optimally sized subgraphs from the input data, en-

abling the efficient handling of complex link prediction tasks. This framework builds upon the SEAL model, enhancing its capabilities by integrating graph structure representation, policy learning, and intelligent subgraph resizing. This combination offers a more precise and resource-efficient solution. Notably, Policy-SEAL distinguishes itself from traditional approaches by its ability to adaptively learn subgraph sizes. Unlike conventional methods that rely on fixed subgraph sizes and may overlook dataset-specific variations, Policy-SEAL leverages reinforcement learning to determine the optimal subgraph size based on each dataset's unique characteristics. This adaptive approach not only enhances accuracy but also leads to savings in computing resources and labor costs.

The core idea of Policy-SEAL is to model the link prediction task as a Markov decision process, where each state represents the information content of the current subgraph and other related information. By utilizing deep Q-learning (DQN) [20], Policy-SEAL can automatically select the optimal subgraph size under different states, enabling the dynamic adjustment of the subgraph size and ultimately enhancing its adaptability and flexibility.

The implementation process of the Policy-SEAL framework is divided into three key parts. First, it leverages a graph structure to represent multi-party conversations, capturing complex interactive relationships more effectively than traditional data representation forms. This approach provides a more intuitive representation, with nodes denoting individuals and edges denoting interactive relationships, thereby improving the accuracy of relationship determination. In the next stage, the framework employs graph neural network-based policy learning to optimize these graphical representations. Through comprehensive analysis and learning, the model identifies and comprehends crucial elements and patterns within conversations, enhancing the accuracy of understanding key elements and contributing to the overall efficacy of the framework. Finally, the learned meta-policy is integrated into the SEAL framework to facilitate its role in MPC reply relationship determination tasks. By applying the meta-policy, the framework's ability to accurately determine relationships and adapt to various dialogue scenarios is strengthened. These three interrelated components of the Policy-SEAL framework work in tandem, establishing a robust foundation for the system. Their synergy ensures optimal performance both in theory and practice. Through this collaborative approach, the Policy-SEAL framework adeptly manages complex MPC tasks, underscoring its considerable potential for migration and practical application.

The Policy-SEAL framework, depicted in Figure 1, effectively determines reply relationships in MPCs using a blend of graph structure representation, policy learning, and dynamic subgraph size adjustment. Sections 4 and 5 provide a detailed explanation of the framework's implementation and assess its performance on real-world datasets.

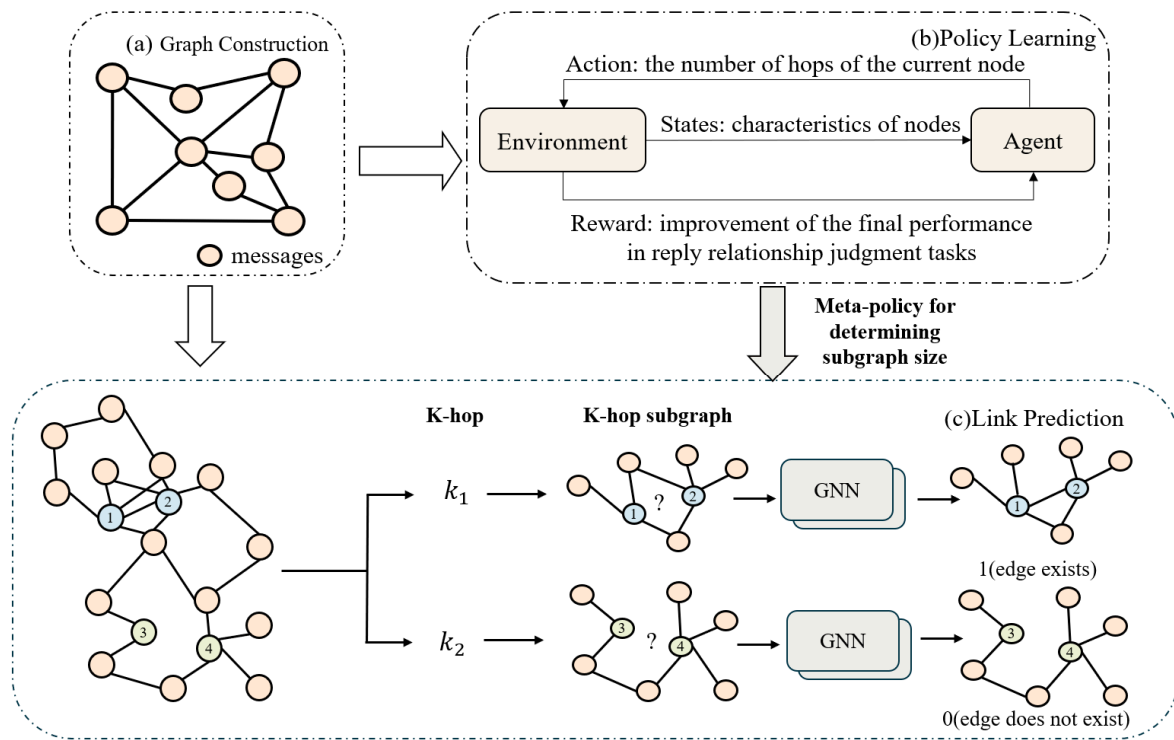


Figure 1. The framework of Policy-SEAL.

4. Graph-Based Model Reply Relationship Identification in Multi-Party Conversations

4.1. Graph Construction

In this section, we describe our methodology for constructing a graph to identify reply relationships in multi-party conversations. In this graph, each utterance is represented as a node. An edge is established between two nodes if there is a reply relationship between the corresponding utterances. This structure allows for an intuitive representation of complex dialogic relationships through the interconnections of nodes and edges. Initially, we collect and preprocess the conversation data, which involves addressing missing values and outliers to ensure data quality. Subsequently, we extract textual, temporal, and interaction features from each utterance. These features facilitate the accurate identification and depiction of reply relationships within the conversations. Directed edges are then established between nodes, with the edge direction indicating the reply flow.

In this study, we employ a graph structure to model the complex interactions among discourses in multi-party conversations (MPCs). We conceptualize these discourses as individual nodes within a unified framework. Initially, a graph $\mathbb{G}(\mathbb{V}, \mathbb{E})$ is constructed based on a given MPC instance comprising M utterances. Here, \mathbb{V} represents the nodes corresponding to the utterances, with each node denoting a separate utterance. The set of directed edges, $\mathbb{E} = \{e_{p,q}\}_{p,q=1}^M$, is established to delineate the connections between the nodes. Each edge $e_{p,q}$ specifies the relationship from node p to node q in the graph. Notably, if there exists a response dynamic between utterance p and utterance q , then edge $e_{p,q}$ is defined to capture this relationship.

In the encoding process, each node is represented as a vector. Initially, a [CLS] token is appended to the start of each utterance, and a [SEP] token is used to denote the end. Subsequently, each sentence is encoded separately using a transformer encoder. The output of each transformer encoder layer is then passed to the subsequent layer. Finally, the discourse representation from the l -lth transformer layer is depicted in Equation (1).

$$H_m^{l+1} = \text{TransformerEncoder}(H_m^l) \tag{1}$$

4.2. Policy Learning

In this study, we employ reinforcement learning techniques to execute policy learning. Initially, we train the meta-policy using the Deep Q-Network (DQN) method. Subsequently, this meta-policy is used to train a graph neural network (GNN). We enhance training efficiency by incorporating a buffer mechanism and sharing parameters, which proves beneficial in practical applications [21].

4.2.1. Reinforcement Learning for Aggregation Iteration

The process of learning the optimal aggregation strategy can be modeled as a Markov decision process (MDP). In this model, node attributes are initially treated as states. Actions are defined by the number of hops, and each action corresponds to a specific subgraph comprising k-hop neighbors. Here, 'k' is determined by a meta-policy output, which also selects a pre-built k-layer GNN architecture from the GNN module depicted on the right side of Figure 2. This architecture is instrumental in generating node representations and capturing reward signals to update the meta-policy. Fundamental elements of the MDP include states, actions, rewards, and state transition probabilities. The following discussion will further delineate these components within the context of graph representation learning.

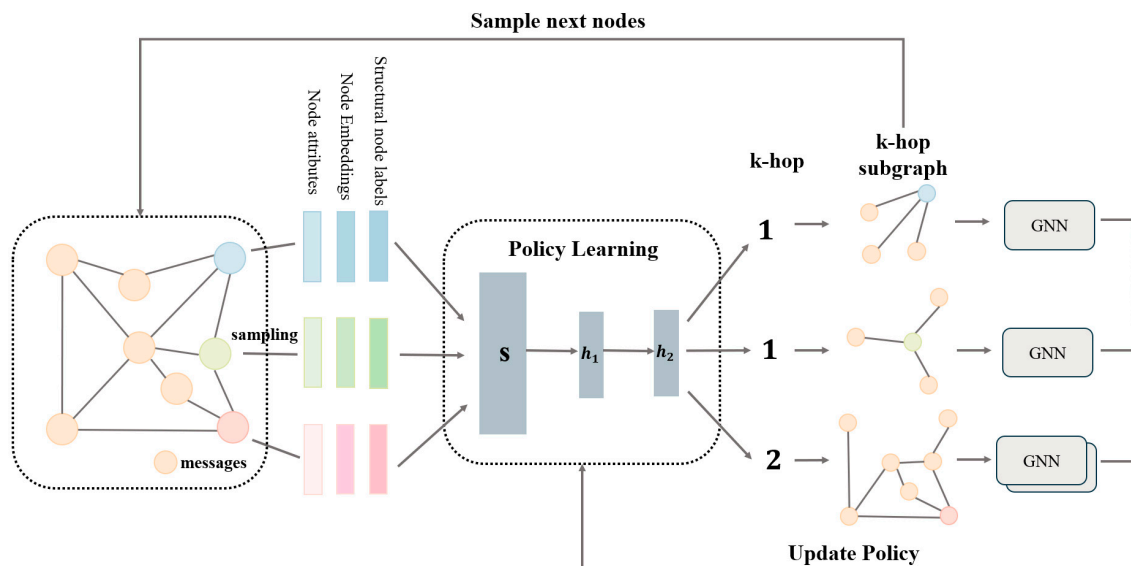


Figure 2. The framework of policy learning modules.

At each time step t , the attributes of the current node (denoted as state) are considered, along with its degree of connectivity, referred to as the number of hops (action). The reward at each time step t corresponds to the improvement in final performance in the link prediction task, capturing the enhancement over the previous state. The aggregation process involves three stages: Initially, the starting node is selected, and its attributes define the current state s_t . Following this, a policy derived from s_t generates an action a_t , which specifies the number of hops for the current node. Subsequently, sampling from the hop neighborhood of the node a_t determines the next state s_{t+1} . Figure 2 provides a demonstration of applying a Markov decision process (MDP) in graph representation learning. For a given action k , the subsequent state is sampled from the k -hop neighborhood. Given the discrete nature of the action space in the aggregation process, where action $a \in \mathcal{A}$ belongs to a finite set of positive integers, this study employs a Deep Q-Network (DQN) for resolution. A key aspect of the DQN is the interpretation of the reward signal, and this article adopts a baseline within the definition of the reward function.

$$\mathcal{R}(s_t, a_t) = \lambda \left(\mathcal{M}(s_t, a_t) - \frac{\sum_{i=t-b}^{t-1} \mathcal{M}(s_i, a_i)}{b-1} \right) \tag{2}$$

4.2.2. Utilizing Strategies for Multi-Node Representation Learning

To demonstrate the effectiveness of the proposed framework, we apply it to a basic graph convolutional network (GCN) [4], learning node representations as outlined in Equation (3). This framework assigns hop numbers to node attributes using specific strategies. These hop numbers are then used to design a custom graph neural network (GNN) architecture for each node, as illustrated in Figure 2. By aggregating information explicitly, the framework ensures both efficiency and adaptability.

$$h_v^k = \sigma\left(\sum_{u \in v \cup N_1(v)} \tilde{A}_{uv} W_k h_u^{k-1}\right) \tag{3}$$

The normalized adjacency matrix of node uv , denoted as \tilde{A}_{uv} , impacts the k th graph convolution layer via the trainable parameter W_k . Here, $N_1(v)$ denotes the first-order neighborhood of node v , and h_v^k signifies the features of node v in the k th layer.

Unlike traditional approaches that utilize a fixed number of layers for each node, the framework introduced in this paper dynamically aggregates a_t layers corresponding to each node attribute s_t at time step t , as opposed to a fixed number of layers for each node. The GNN architecture is defined by the following transfer equation.

$$\begin{aligned} h_v^1 &= \sigma\left(\sum_{u_1 \in \{v\} \cup N_1(v)} \tilde{A}_{u_1 v} W_1 X_u\right) \\ &\quad \vdots \\ h_v^{k=a_t} &= \sigma\left(\sum_{u_k \in u_{k-1} \cup N_k(v)} \tilde{A}_{u_1 u_{k-1}} W_k h_v^{k-1}\right) \\ \text{output} &= \text{softmax}\left(h_v^{k=a_t}\right) \end{aligned} \tag{4}$$

At time step t , the aggregation number $k = a_t$ is determined by the $\tilde{\pi}$ function. The feature representation of node v in the k th layer is denoted as h_v^k , while the input attribute vector of node u is represented as X_u . The trainable parameter in the k th graph convolution layer is denoted as W_k , and the activation function of each layer is represented by σ .

4.2.3. Improving GNN Training Policy with Parameter Sharing and Buffering Mechanism

To enhance training efficiency, this paper introduces a parameter sharing mechanism designed to reduce the number of training parameters required, thus minimizing the time cost associated with reconstructing the graph neural network (GNN) at each time step. Additionally, a buffering mechanism is employed to boost efficiency when training multiple GNNs for distinct actions concurrently. Our approach utilizes parameter sharing to decrease training parameters, subsequently reducing the reconstruction time for the GNN each time step. Concurrently, a buffer mechanism is employed to augment training efficiency. When the buffer reaches the designated batch size, selected layers are utilized to train the GNN, after which the buffer is cleared. This method also enhances the optimization of the Q function through the random sampling of batch data from the buffer, thereby improving model performance.

The implementation begins by establishing the maximum number of GNN layers. At each time step t , layers are stacked in sequence according to the action a_t to effectively train the GNN. Node attributes and actions are stored in the action buffer, with periodic checks to ascertain if the buffer has reached its batch size capacity. Upon reaching capacity, the buffer's contents are used to construct and train a GNN with the selected layers, and then the buffer is cleared. This strategy improves training efficiency compared to training a single GNN at each time step. This paper leverages random sampling from the memory buffer to utilize experiences stored therein, optimizing the Q function for subsequent iterations as delineated in Formula (5).

$$Q(s, a) = \mathbb{E}_{s'}[\mathcal{R}(s') + \gamma \max_{a'} Q(s', a')] \tag{5}$$

$Q(s, a)$ represents the expected return of taking action a in state s , s' is the next state, and a' is the next action. $\mathcal{R}(s')$ is the immediate reward fed back to the agent by the environment, and γ is the discount factor, whose value is between 0 and 1 and is used to calculate the present value of future rewards. $\max_{a'} Q(s', a')$ represents the maximum expected return that the agent can obtain in the next state s' .

4.3. Link Prediction

We employ the SEAL framework as the basis for policy learning, which allows us to dynamically determine the size of the extracted subgraph. This method entails extracting the closure from both sampled positive links (observed) and sampled negative links (unobserved) within the subgraph to create the training data.

In the Policy-SEAL system, the node information matrix comprises structural node labels, node embeddings, and node attributes. Structural node labels are assigned to nodes in the closed subgraph to indicate their relative positions to the target node, thereby capturing the structural characteristics of the nodes. A node labeling function $f_l : V \rightarrow \mathbb{N}$ is employed to assign each node i in the closed subgraph an integer label $f_l(i)$, which serves to differentiate the various roles of nodes within the closed subgraph.

5. Experiments

This section will introduce the dataset, describe the comparative and ablation experiments, outline the experimental settings, and present the final results to thoroughly analyze our research findings.

5.1. Datasets

We utilized the publicly available Ubuntu IRC dataset [22] for our experiments due to its detailed records of multi-party conversations, which resemble the intricate communication patterns within cyber-physical systems (CPSs). This dataset includes comprehensive records specifying the identity of participants and the content of their messages, making it an excellent proxy for evaluating the efficacy of our proposed method in identifying reply relationships within CPS environments. By benchmarking our results against existing studies that also used this dataset, we ensure the robustness and applicability of our approach to real-world CPS communications. Our method demonstrated superior performance across various evaluation metrics compared to existing techniques. Specifically, in the multi-party conversation (MPC) task, we evaluated the performance of Policy-SEAL on the Ubuntu IRC benchmark introduced by Hu et al. This benchmark provides speaker and receiver labels for each utterance and consists of training, validation, and test sets containing 311,725, 5000, and 5000 conversations, respectively.

To further demonstrate the versatility of our method across different link prediction scenarios, we conducted experiments on four datasets from the Open Graph Benchmark (OGB) [23]: ogbl-ppa, ogbl-collab, ogbl-ddi, and ogbl-citation2. These datasets are open source, large-scale (featuring up to 2.9 million nodes and 30.6 million edges), and follow standard evaluation protocols, making them ideal benchmarks for assessing the real-world performance of link prediction algorithms. For detailed statistics on the OGB datasets, see Table 1.

Table 1. Information of Open Graph Benchmark.

Dataset	Number of Nodes	Number of Edges	Train/Validation/Test	Evaluation
ogbl-ppa	576,289	30,326,273	70/20/10	Hits@100
ogbl-collab	235,868	1,285,465	92/4/4	Hits@50
ogbl-ddi	4267	1,334,889	80/10/10	Hits@20
ogbl-citation2	2,927,963	30,561,187	99/1/1	MRR

5.2. Comparative and Ablation Experiments

To assess the efficacy of Policy-SEAL in analyzing the relationships between messages, we conducted experiments on the Ubuntu IRC dataset. We also performed comparative and ablation studies with alternative models to validate our method's efficiency. The following representation learning algorithms were selected for comparison:

1. TextCNN [24]: The Convolutional Neural Network for Sentence Classification (TextCNN) is widely used in text classification tasks to capture local correlation features within text. Separate feature vectors are obtained for two messages using TextCNN. These vectors are then combined and transformed into an objective function to determine the reply relationship between the messages.
2. BERT [25]: This involves utilizing the pre-trained BERT model to initialize the vector representations. BERT is a robust natural language processing model widely used in various NLP tasks.
3. ESIM [26]: This algorithm employs Bi-LSTM and attention mechanisms for sequential reasoning, merging local and global reasoning capabilities.

BERT, TextCNN, and ESIM were chosen as comparison models due to their strong performance in natural language processing and conversation analysis tasks. Additionally, models such as GraphSAGE and the GCN were considered to further substantiate the advantages of our approach. For the baseline methods on the OGB, we used GraphSAGE [27], the GCN [4], and GCN + DRNL [15], all of which utilize pairwise node representations generated by GNNs for link representation. Specifically, GCN + DRNL employs the same node labeling approach as Policy-SEAL discussed in this paper. In the ablation study, we primarily assessed the experimental outcomes by excluding the policy learning submodule.

5.3. Setup

In this study, we employ both graph convolutional network (GCN) and Deep Q-Network (DQN) methodologies. The GCN layer utilizes the Rectified Linear Unit (ReLU) as the activation function and includes a dropout mechanism with a rate of 0.5 to prevent overfitting. The model optimization is performed using the Adam optimizer, set with a learning rate of 0.01 and a weight decay of 0.0005. We maintain a batch size of 128 for training purposes. The DQN approach is implemented through the RLCard framework [28]. For the construction of the Multi-Layer Perceptron (MLP), the network comprises five layers with hidden units configured as 32, 64, 128, 64, and 32, respectively. Furthermore, the memory buffer is sized at 10,000 units.

5.4. Results

5.4.1. Comparative Experiment Results

In this section, we evaluate the effects of the model and several comparison models on two tasks: (1) assessing the reply relationship between messages and (2) predicting links on the Ubuntu IRC and the OGB datasets, respectively. We collect statistics on the validation sets of both datasets to analyze the experimental results.

(1) Comparative experiment on Ubuntu IRC

To present the experimental data of each model more clearly, we computed the accuracy for all models and compared their performance on the Ubuntu IRC dataset. The optimal results are in bold in Table 2, which also includes statistical summaries. In this study, we investigated the effects of employing a fixed versus a flexible subgraph hop count extraction through the Policy-SEAL strategy to analyze message-passing conversation (MPC) reply relationships. Moreover, we evaluated the impact of different hop count settings on model accuracy and computational efficiency, utilizing a 3090 GPU. These experimental outcomes are displayed in Table 3, accompanied by relevant statistical data.

Table 2. Experimental results on Ubuntu IRC.

Ubuntu IRC	BERT	TextCNN	ESIM	Policy-SEAL
Acc. on val (%)	69.51 ± 0.12	56.59 ± 0.52	56.06 ± 1.26	81.63 ± 0.35
Acc. on test (%)	64.48 ± 0.35	54.59 ± 0.39	51.91 ± 1.82	80.57 ± 0.16

Table 3. Experimental results of Fixed-Hop and Policy-SEAL.

Hop Count	Acc. on Val (%)	Acc. on Test (%)	Efficiency
6	68.72 ± 1.54	65.41 ± 1.86	292 min
8	75.43 ± 0.94	74.17 ± 1.06	503 min
10	78.58 ± 0.79	78.41 ± 0.57	716 min
12	82.15 ± 0.41	81.09 ± 0.32	1039 min
Policy-SEAL	81.63 ± 0.35	80.57 ± 0.16	784 min

(2) Comparative experiment on OGB

The experimental results of the proposed method were compared with other established link prediction techniques using four datasets from the Open Graph Benchmark (OGB) specifically designed for link prediction. The results are presented in Table 4, where the best-performing outcomes are in bold to enhance visibility. This table provides a comprehensive overview of the performance metrics for each model.

Table 4. Experimental results on OGB.

Dataset (Evaluation)	GraphSAGE		GCN		GCN+DRNL		Policy-SEAL	
	Validation	Test	Validation	Test	Validation	Test	Validation	Test
ogbl-ppa hits@100 (%)	17.24 ± 2.64	16.55 ± 2.40	18.85 ± 1.40	18.67 ± 1.32	46.43 ± 3.03	45.24 ± 3.95	52.31 ± 1.92	49.98 ± 2.12
ogbl-collab hits@50 (%)	56.88 ± 0.77	54.63 ± 1.12	52.63 ± 1.15	47.14 ± 1.45	64.51 ± 0.43	64.40 ± 0.45	68.91 ± 0.14	61.01 ± 0.40
ogbl-ddi hits@20 (%)	62.62 ± 0.37	53.90 ± 4.47	55.50 ± 2.08	37.07 ± 5.07	29.47 ± 1.54	22.81 ± 4.93	30.21 ± 2.27	32.64 ± 2.74
ogbl-citation2 MRR (%)	82.63 ± 0.23	82.60 ± 0.36	84.79 ± 0.23	84.74 ± 0.21	81.07 ± 0.30	81.27 ± 0.31	89.09 ± 0.35	87.98 ± 0.52

(3) Analysis of experimental results

Table 2 presents a comparison of the performance of our proposed method against various baseline methods using the Ubuntu IRC dataset. The analysis indicates that Policy-SEAL outperforms the baselines on both the validation and test sets, demonstrating its effectiveness in assessing reply relationships in multi-party conversations (MPCs). Unlike the baseline model, which primarily relies on text content and subgraph structures of message pairs, our model additionally incorporates contextual cues and detailed structural information. This comprehensive approach enhances the model's understanding of the data and highlights the importance of structural insights in evaluating message reply relationships.

The analysis of Table 3 shows that the model's predictive accuracy on the Ubuntu IRC dataset improves as the number of fixed hops increases. However, this improvement comes with a significant increase in runtime. This suggests that our method balances efficiency and effectiveness. In contrast, Policy-SEAL achieves comparable predictive performance to high hop count strategies but maintains a lower runtime, thus demonstrating its efficiency. This underscores the effectiveness of Policy-SEAL's flexible subgraph extraction method in improving the assessment of MPC reply relationships.

The ogbl-ddi dataset, characterized by its high density with 4267 nodes and 1,334,889 edges (average node degree of 500.5), presents challenges for inductive learning in graph neural networks (GNNs), as it hampers their ability to discern meaningful structural patterns. Table 4 shows that while Policy-SEAL performs well across the three datasets from the Open Graph Benchmark (OGB), its results on ogbl-ddi are comparatively weaker. Implementing trainable node embedding strategies, similar to those used in graph autoencoders (GAEs), shifts the focus from inductive pattern learning to node embeddings. This shift potentially improves performance on densely connected datasets like ogbl-ddi by enhancing the model's versatility and adaptability to different data structures.

5.4.2. Results of Ablation Experiments

In this section, we explore the impact of the policy learning submodule on the link prediction task and the MPC reply relationship judgment task. The results from the ablation study are presented in Table 5, where the best results are indicated in bold. These results indicate that the original model achieves the best performance on the dataset. This confirms that the strategy of incorporating policy learning significantly enhances the capabilities of the model in both link prediction and reply relationship judgment tasks.

Table 5. The ablation experiment results.

Datasets (Evaluation)	SEAL		Policy-SEAL	
	Validation	Test	Validation	Test
ogbl-ppa Hits@100 (%)	51.25 ± 2.52	48.80 ± 3.16	52.31 ± 1.92	49.98 ± 2.12
ogbl-collab Hits@50 (%)	64.95 ± 0.43	54.71 ± 0.49	68.91 ± 0.14	61.01 ± 0.40
ogbl-ddi Hits@20 (%)	28.49 ± 2.69	30.56 ± 3.86	30.21 ± 2.27	32.64 ± 2.74
ogbl-citation2 MRR (%)	87.57 ± 0.31	87.67 ± 0.32	89.09 ± 0.35	87.98 ± 0.52
Ubuntu IRC ACC (%)	78.93 ± 0.15	77.78 ± 0.22	81.63 ± 0.07	80.57 ± 0.16

In all experimental datasets, Table 5 demonstrates a decline in performance during ablation experiments that exclude the policy learning submodule. The most significant decline is observed in the ogbl-collab dataset, with decreases of 3.96% and 6.30%, respectively. This decline can be attributed to the role of the original model's policy learning submodule, which decodes the structural relationships within graphs and autonomously determines the optimal subgraph size based on the unique structural characteristics of various graphs. These results confirm the effectiveness of the policy learning submodule.

6. Conclusions

In response to malicious tampering in device interactions within CPSs, such as alterations to content or structure intended to mislead system judgment and disrupt accurate communication link identification, we developed a novel policy search method, Policy-SEAL. This method uses reinforcement learning to adaptively determine the optimal subgraph size, enhancing the accuracy of communication link assessments in CPS interactions. Policy-SEAL dynamically adjusts the subgraph size based on the unique characteristics of the interaction dataset, generating more precise graph representations and significantly boosting the accuracy of communication link judgments. Furthermore, experiments demonstrate the substantial effectiveness of Policy-SEAL in managing CPS communication link judgment tasks. The method not only achieved exceptional accuracy

on the Ubuntu IRC dataset but also displayed broad applicability and portability across other link prediction tasks.

The experimental outcomes underscore the remarkable performance enhancement of Policy-SEAL compared to baseline models and validate the effectiveness of the introduced policy search technique. By optimizing subgraph size strategies during the validation phase, Policy-SEAL automatically identifies optimal subgraph representations for different datasets, achieving superior results in subsequent tasks. Ultimately, this method offers a versatile and comprehensive approach to acquiring graph representation skills, transforming the MPC reply relationship judgment task into a link prediction challenge, and utilizing graph learning technology to capture the implicit relationships between messages—thereby saving computational resources and reducing labor costs effectively.

Despite these significant advances, this study has limitations, such as the small size of the experimental dataset, which may not cover all potential CPS application scenarios comprehensively. Additionally, the model's high complexity and computing resource demands may require further optimization for real-world deployment. Future research should aim to address these challenges and explore broader application scenarios.

Author Contributions: The authors confirm their contributions to this article as follows: research conception design: B.Z. and B.H.; data collection: W.S.; analysis and interpretation research results: J.Y. and B.H.; manuscript preparation: L.Q. and B.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Natural Science Foundation of China (Nos. 62072488), partially supported by Beijing Natural Science Foundation of China (Nos. 4202064).

Data Availability Statement: The two data used in this study can be found in references [22] and [23], respectively.

Acknowledgments: We are grateful to Lirong Qiu for her careful guidance and assistance. Our sincere appreciation goes to the research team for their teamwork. We also extend special thanks to the reviewers for their enthusiastic suggestions.

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

References

1. Zhang, R.; Lee, H.; Polymenakos, L.; Radev, D. Addressee and Response Selection in Multi-Party Conversations with Speaker Interaction rnns. In Proceedings of the AAAI Conference on Artificial Intelligence, New Orleans, LO, USA, 2–7 February 2018; pp. 5690–5697.
2. Presekal, A.; Ştefanov, A.; Rajkumar, V.S.; Palensky, P. Cyber-physical power system attack graph model based on hybrid deep learning. *IEEE Trans. Smart Grid* **2023**, *14*, 4007–4020. [CrossRef]
3. Wu, W.; Song, C.; Zhao, J.; Xu, Z. Physics-informed gated recurrent graph attention unit network for anomaly detection in industrial cyber-physical systems. *Inf. Sci.* **2023**, *629*, 618–633. [CrossRef]
4. Kipf, T.N.; Welling, M. Semi-Supervised Classification with Graph Convolutional Networks. In Proceedings of the 5th International Conference on Learning Representations (ICLR), Toulon, France, 6 February 2017. Available online: <https://openreview.net/forum?id=SJU4ayYgl> (accessed on 9 September 2016).
5. Velickovic, P.; Cucurull, G.; Casanova, A.; Romero, A.; Lio, P.; Bengio, Y. Graph Attention Networks. *arXiv* **2017**, *1050*, 48550.
6. Kipf, T.N.; Welling, M. Variational graph auto-encoders. *arXiv* **2016**, arXiv:1611.07308.
7. Wei, J.; Shuster, K.; Szlam, A.; Weston, J.; Urbanek, J.; Komeili, M. Multi-Party Chat: Conversational Agents in Group Settings with Humans and Models. *arXiv* **2023**, arXiv:2304.13835. Available online: <https://arxiv.org/abs/2304.13835v3> (accessed on 6 April 2023).
8. Gu, J.-C.; Tao, C.; Ling, Z.-H. Who Says What to Whom: A Survey of Multi-Party Conversations. In Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence Survey Track, Vienna, Austria, 23–29 July 2022; pp. 5486–5493. [CrossRef]
9. Ouchi, H.; Tsuboi, Y. Addressee and Response Selection for Multi-Party Conversation. In Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, Austin, TX, USA, 1–5 November 2016; pp. 2133–2143.
10. Zhu, H.; Nan, F.; Wang, Z.; Nallapati, R.; Xiang, B. Who Did they Respond to? Conversation Structure Modeling using Masked Hierarchical Transformer. In Proceedings of the AAAI Conference on Artificial Intelligence, New York, NY, USA, 7–12 February 2020; Volume 34, pp. 9741–9748.

11. Liu, H.; Shi, Z.; Zhu, X. Unsupervised Conversation Disentanglement through Co-Training. In Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, Online, Punta Cana, Dominican Republic, 7–11 November 2021; pp. 2345–2356.
12. Ma, X.; Zhang, Z.S.; Zhao, H. Structural characterization for dialogue disentanglement. In Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Dublin, Ireland, 22–27 May 2022; pp. 285–297.
13. Yu, N.; Fu, G.; Zhang, M. Speaker-aware discourse parsing on multi-party dialogues. In Proceedings of the 29th International Conference on Computational Linguistics, Gyeongju, Republic of Korea, 12–17 October 2022; pp. 5372–5382.
14. Gu, J.-C.; Tan, C.-H.; Tao, C.; Ling, Z.-H.; Hu, H.; Geng, X.; Jiang, D. HeterMPC: A heterogeneous graph neural network for response generation in multi-party conversations. In Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Dublin, Ireland, 22–27 May 2022; pp. 5086–5097.
15. Elhanashi, A.; Dini, P.; Saponara, S.; Zheng, Q. Integration of Deep Learning into the IoT: A Survey of Techniques and Challenges for Real-World Applications. *Electronics* **2023**, *12*, 4925. [[CrossRef](#)]
16. Zhang, M.; Chen, Y. Link prediction based on graph neural networks. In Proceedings of the NIPS'18: Proceedings of the 32nd International Conference on Neural Information Processing Systems, Montreal, QC, Canada, 3–8 December 2018; pp. 5171–5181.
17. Zhang, M.; Li, P.; Xia, Y.; Wang, K.; Jin, L. Labeling trick: A theory of using graph neural networks for multi-node representation learning. *Adv. Neural Inf. Process. Syst.* **2021**, *34*, 9061–9073.
18. Chen, X.; Zhao, P.; Yu, P.; Liu, B.; Li, W.; Xie, Y.; Yuan, M. Risk analysis and optimization for communication transmission link interruption in Smart Grid cyber-physical system. *Int. J. Distrib. Sens. Netw.* **2018**, *14*, 6035. [[CrossRef](#)]
19. Marashi, K.; Sarvestani, S.S.; Hurson, A.R. Identification of interdependencies and prediction of fault propagation for cyber-physical systems. *Reliab. Eng. Syst. Saf.* **2021**, *215*, 107787. [[CrossRef](#)]
20. Mnih, V.; Kavukcuoglu, K.; Silver, D.; Rusu, A.A.; Veness, J.; Bellemare, M.G.; Graves, A.; Riedmiller, M.; Fidjeland, A.K.; Ostrovski, G.; et al. Human-level control through deep reinforcement learning. *Nature* **2015**, *518*, 529–533. [[CrossRef](#)] [[PubMed](#)]
21. Hu, W.; Chan, Z.; Liu, B.; Zhao, D.; Ma, J.; Yan, R. Policy-GNN: Aggregation optimization for graph neural networks. In Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, Virtual Event, CA, USA, 6–10 July 2020; pp. 461–471.
22. Hu, W.; Chan, Z.; Liu, B.; Zhao, D.; Ma, J.; Yan, R. GSN: A graph-structured network for multi-party dialogues. In Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, Macao, China, 10–16 August 2019; pp. 5010–5016.
23. Hu, W.; Fey, M.; Zitnik, M.; Dong, Y.; Ren, H.; Liu, B.; Leskovec, J. Open graph benchmark: Datasets for machine learning on graphs. *Adv. Neural Inf. Process. Syst.* **2020**, *33*, 22118–22133.
24. Chen, Y. Convolutional Neural Network for Sentence Classification. Master's Thesis, University of Waterloo, Waterloo, ON, Canada, 2015.
25. Devlin, J.; Chang, M.-W.; Lee, K.; Toutanova, K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), Minneapolis, MI, USA, 2–7 June 2019; pp. 4171–4186.
26. Chen, Q.; Zhu, X.D.; Ling, Z.H.; Wei, S.; Jiang, H.; Inkpen, D. Enhanced LSTM for natural language inference. In Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Vancouver, BC, Canada, 30 July–4 August 2017; pp. 1657–1668.
27. Hamilton, W.; Ying, Z.T.; Leskovec, J. Inductive representation learning on large graphs. *Proc. Adv. Neural Inf. Process. Syst.* **2017**, *30*, 1025–1035.
28. Zha, D.; Lai, K.H.; Cao, Y.; Huang, S.; Wei, R.; Guo, J.; Hu, X. RLCARD: A toolkit for reinforcement learning in card games. *arXiv* **2019**, arXiv:1910.04376.

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