Underground Utility Network Completion based on Spatial Contextual Information of Ground Facilities and Utility Anchor Points using Graph Neural Networks

Yuxi Zhang[1] and Hubo Cai[1]

[1]Lyles School of Civil Engineering, Purdue University, United States
zhan2889@purdue.edu, hubocai@purdue.edu

Abstract –

Every year, accidental damage during excavation leads to numerous disruptions in utility services. These incidents cause not only financial losses but also injuries and fatalities. A major contributing factor to these incidents is the lack of accurate location data for utilities. The current practice involves a time-consuming coordination process of obtaining utility maps from owners and field surveys, which is often hindered by delays and incomplete records. In response to these challenges, this paper proposes a novel method to predict underground utility lines in situations where records are unavailable or delayed. Our approach leverages visible utility anchor points, such as manholes, and the spatial context provided by nearby ground features like roads. The methodology involves three primary steps: constructing a relational data model of the utility network, transforming this data into graphs, and employing a graph neural network for prediction. This innovative approach demonstrates good performance, achieving a ROC AUC score of 95.24% in predicting sewer line connections between manholes. This method automates the inference of utility lines, providing utility owners and excavation contractors a solution for identifying unknown connections and reducing risks from inaccurate information.

Keywords –

Underground Utility Network Completion; Spatial Contextual Information; Graph Neural Networks

1 Introduction

The ongoing issue of inaccurate and incomplete information of buried utilities poses a significant challenge across the United States. Annually, numerous utility disruptions are caused by accidental excavation damage. These incidents impact communities and businesses, leading to injuries and tragically resulting in loss of life. According to Common Ground Alliance Damage Information Reporting Tool [1], 87.84% of these incidents occur due to missing or inaccurate location information. Current practice to mitigate these risks involves coordinating with utility owners to access utility maps and employ utility surveyors. The utility map serves as a crucial starting point, providing approximate line locations for further utility surveys. However, obtaining utility records faces prolonged delays in the coordination process, and some records may be entirely absent. Therefore, there's an urgent need to propose a method for inferring utility line locations when utility records are delayed or unavailable.

When records are inaccessible, inferring some utility lines is possible by examining visible utility anchor points like manholes and nearby ground facilities such as roads and buildings. These visible features imply the presence and general locations of utilities. Acquiring information about these visible features is feasible through field surveys or high-resolution satellite imagery. However, this inference relies on scarce professional judgment and expertise, which can be time-consuming, error-prone, and may further complicate the process.

This paper introduces a novel approach for automatically completing underground pipeline networks. It focuses on predicting utility line segments by using visible utility anchor points and ground facilities as spatial contextual cues. The objective is to aid users in inferring the existence and approximate locations of utility lines when utility records are not accessible.

2 Literature Review

2.1 Utility Parameters, Spatial Contexts, and Design Practices for Predicting Utilities

Existing studies [2–6] address the design and completion of utility networks by predicting the presence of pipelines based on their endpoints, such as manholes, and assessing the extent to which the network conforms

to design criteria and practices. For instance, Afshar [2] suggested minimizing the cost function related to pipe diameters and excavation depths while adhering to certain constraints to reflect design criteria. He compiled a list of sewer design practices as constraints, such as the minimum flow velocity required to prevent sediment buildup and the minimum pipe slope necessary to avoid adverse slopes due to inaccurate construction. Similarly, Izquierdo [3] formulated the problem for hydraulic systems, akin to Afshar's approach, but also incorporated the continuity and energy equations of hydraulics into the model. These studies concentrate on using the parameters of the pipeline and pipe endpoints to aid in the design of the pipeline network.

Furthermore, some research extends beyond the parameters of pipes and their endpoints. It also examines the relationship of these endpoints to visible ground elements and the surrounding environment, such as catchments, roads, and buildings near the pipes. For example, Bailly et al. [4] predicted the presence of pipelines based on the cumulative length of pipelines in relation to the catchment extent and network connectivity. Chahinian et al. [7] used manhole locations and elevations to predict the presence of pipelines and minimize instances of lines intersecting buildings and roads. Their top result achieved a precision and recall of 0.92 each, alongside a critical success index of 0.85. These studies underscore the importance of spatial contexts in enhancing pipeline network predictions.

2.2 Challenges and Limitations in Existing Studies

Existing studies carefully consider the information crucial for completing or designing pipeline networks. However, they face challenges in both mathematically modeling and solving the problem as follows:

1. One primary difficulty is the unknown correlation among pipe endpoint parameters, their connection parameters, spatial contexts, and pipeline presence. Existing studies simplify the problem by assumptions, leading to a lack of justification.
2. Another challenge is capturing the interdependency of variables within a network solely through human knowledge. This limits current methods to focusing only on parameters directly connected to the pipelines or nearby ground facilities, overlooking broader interdependencies.
3. Additionally, even when correlations and relationships are simplified and mathematically formulated, solving the model becomes computational expensive. These problems are often approached as combinatorial optimization, aiming to minimize costs while considering various constraints. The complexity of these problems is compounded by non-linear functions and constraints, resulting in a solution space filled with numerous local minima and discontinuities. Consequently, studies have resort to computational expensive optimization methods such as heuristic algorithms, particle swarm, ant colony optimization, and others, in pursuit of the global optimal solution.

In summary, current research mainly utilizes rule-based approaches to predict pipeline connections between two endpoints, considering both their parameters and spatial contexts. This body of research highlights the complexities involved in formulating and solving these problems, especially the challenges in converting industry practices into effective cost functions. It indicates that explicitly modeling this problem relying solely on human knowledge presents significant challenges. Additionally, the complexities hinder further exploration of factors, such as the detailed spatial relationships between manholes and their surrounding environment, related to pipeline prediction.

2.3 Advantages of GNNs in Pipeline Network Completion

In the context of pattern recognition, learning-based methods can overcome the limitations of previous studies that struggled with explicitly modeling cost functions. With sufficient data, machine learning can quickly adapt to data from diverse practices.

Among the learning-based approaches, Deep Neural Networks (DNNs) [8,9] distinguish themselves from traditional machine learning methods by simultaneously learning features and objective functions. The advantages of using it for this problem lie in three aspects:

1. Alignment with Graphical Data Structures: Pipeline networks are inherently structured in a graphical format, with manholes serving as nodes and pipelines as edges. This naturally aligns with the architecture of Graph Neural Networks (GNNs), facilitating the integration of information into a unified network for discerning data correlations. Additionally, this problem can be formulated as linkage prediction in GNN studies [10], a well-established research area that is supported by a solid mathematical and statistical foundation.
2. Feature Extraction from Subgraphs: GNNs specialize in handling graph-structured data, enabling the extraction of comprehensive features from subgraphs [11]. These methods, known as graph embeddings, allow for the representation of pipeline networks by aggregating information not just from direct connections but also from the broader network context.
3. Discriminative Feature Learning: The concurrent learning of features and objective functions lead to

learn discriminative implicit representations [8]. In contrast to traditional methods that linearly model relationships between handcrafted features, DNNs excel in learning implicit feature representations that encode complex relationships within the data. These features are specifically optimized for the downstream task, enhancing the accuracy and effectiveness of pipeline network predictions [12].

4. Recent advances in using GNN models in Geospatial Artificial Intelligence (GeoAI): GNN models are particularly adept at handling geospatial challenges that involve analyzing points of interest, their spatial relationships, and non-grid topologies. GNNs have shown notable effectiveness in applications such as traffic flow [13] and PM2.5 level forecasting [14], where training and testing occur on the same nodes, referring to transductive learning. A significant challenge in GeoAI, however, is the application of models trained on one set of location data to completely new, unseen locations, known as inductive learning. To address this, significant advancements have been made in geospatial location encoding techniques [15]. These techniques transform location data, whether two- or three-dimensional, into a high-dimensional feature vector. This approach preserves relative distances and, optionally, directional relationships between locations, enhancing the model's ability to adapt to new locations not seen in the training phase.

Therefore, there is a need to explore the potential to overcome the limitations of current utility network completion methods using data-driven approaches. This study mainly introduces the framework of utility line prediction, addressing the following two challenges: (1) identifying which spatial and semantic contexts to include along with their encoding techniques; and (2) designing GNN models capable of efficiently propagating information across a heterogeneous graph—such as nodes representing manholes and roads—and learn features for network topology prediction.

3 Methodology

The overall framework is illustrated in Figure 1. The process begins with building a relational data model to organize information on utility anchor points, lines, and ground facilities and their spatial relationships. Second, all the records in the relational data model are represented as graphs, with anchor points and facilities as nodes, and utility lines and their relationships as edges. Finally, a GNN model is developed to predict utility lines, which are the links between anchor point nodes.

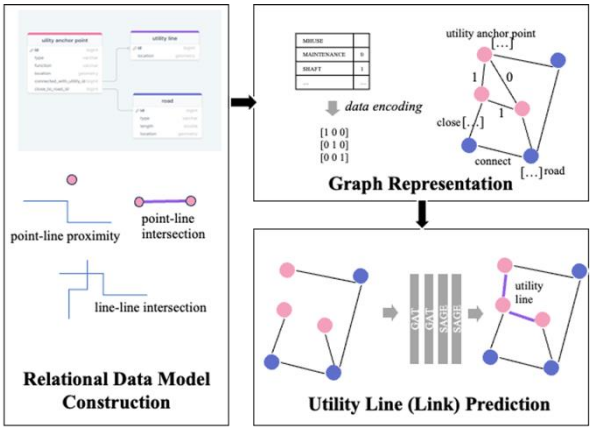


Figure 1. Overall Framework

3.1 Relational Data Model Construction

Geospatial relational data modeling is a crucial step to present the properties and the relationships among different entities. It not only facilitates data extraction from existing databases but also aids in building the graph representations of the utility anchor points, lines and ground facilities. Figure 2 depicts the Entity-Relationship (ER) diagram.

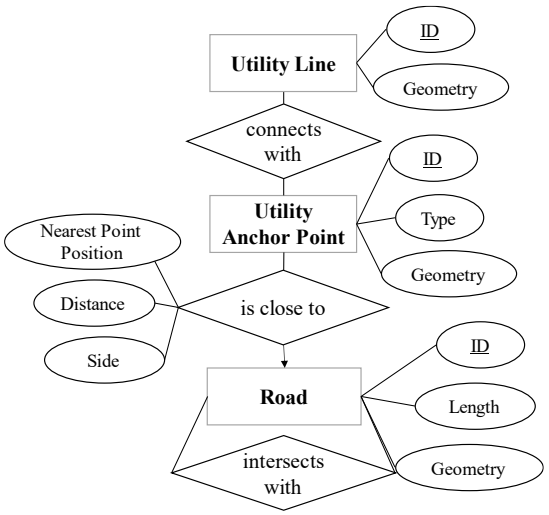


Figure 2. Entity-Relationship Diagram

In this diagram, three entities are used:

1. Utility Anchor Point: These are visible utility line junctions, such as manholes and ground pumps, indicating the locations of underground lines. Attributes include ID, type, and geometry.
2. Road: Roads, as a typical ground facility, provide spatial contextual cues for utility line prediction. The alignment of utility lines along roads makes this data a potential indicator. Additionally, roads are accessible from satellite imagery and digital

road maps, which are widely available. Roads are characterized by their ID, length, and geometry.

3. Utility Line. These are typically buried utility lines that are the focus of prediction in this research. Data on these lines are used for model training and for validation and testing in the evaluation stage. As this study focuses on predicting the existence of lines, only ID and geometry information are utilized.

Three relationships are established based on spatial relationship analysis:

1. Utility Line-Anchor Point Connection: The connection between utility lines and anchor points is determined by merging two tables through point-line intersection analysis.
2. Anchor Point-Road Proximity: Anchor point-road proximity is identified by locating the nearest line to the anchor point, considering only those within a 100-meters radius as "close." Additionally, three attributes are extracted: the position of the nearest point on the road, the distance from the anchor point to this nearest road point, and the side of the road on which the anchor point is located. These attributes aid in predicting utility line placement, as most lines run parallel to, rather than across, roads. For instance, two connected manholes are likely to be on the same side of the road and in proximity.
3. Road-Road Intersection: The road-road relationship is built by merging road tables through line-line intersection analysis.

3.2 Graph Representation

Building the graph representation of the utility network and its surroundings, based on the geospatial data model, involves three main steps: (1) establishing relationships between anchor points through their connections with utility lines; (2) converting the relational data model into a graph data model; (3) encoding the data with numerical values.

3.2.1 Anchor Point to Anchor Point Relationship Establishment

This step transforms the utility line entity into relationships between anchor points. It is designed to align with the objective of predicting utility lines, which will be modeled as the edges between anchor point nodes in the graph network. Typically, in the graph data model, edges represent relationships in the relational data model.

The implementation process is straightforward. A list of anchor-point ID pairs is generated if they intersect with the same utility line segments. This action removes the utility line entity in the relational data model and establishes a many-to-many relationship between the anchor points themselves.

3.2.2 Relational Data Model to Graph Data Model Conversion

This step follows the typical process of transforming the relational database to graph database, including the following steps: (1) table to node label; (2) row to node; (2) column to node property; (3) foreign key to edge; (4) relationship attributes to edge properties.

Anchor Point Entity Table

	ASSETID	SUBTYPECD	geometry
105060	MH198430	22201MAINT	POINT (152.94858 -27.44766)
104560	MH198431	22201MAINT	POINT (152.94826 -27.44747)

Anchor Point – Anchor Point Relation Table

	MH_index1	MH_index2	line_index
1	105060.0	104560.0	2.0

Anchor Point – Road Relation Table

	index_MH	index_Road	NEAR_POS	SIDE	dist_bins
125555	104560	21457	0.443777	1.0	4
125558	105060	21457	0.385538	1.0	4

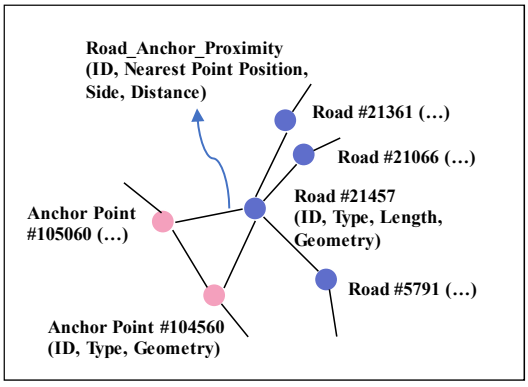
Road – Road Relation Table

	road_index1	road_index2
17467	21457	5791
63160	21457	21066
64063	21457	21361

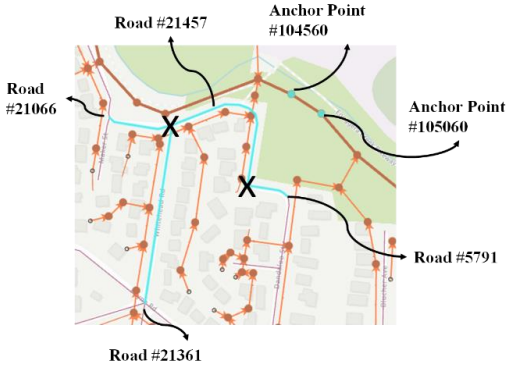
Road Entity Table

	ROUTE_TYPE	Shape_Leng	geometry	FID
21457	Neighbourhood / local	170.465693	LINESTRING (494839.784 6963903.639, 494850.033...	26755
5791	Neighbourhood / local	43.593375	LINESTRING (494839.784 6963903.639, 494873.089...	7351
21066	Neighbourhood / local	68.886734	LINESTRING (494760.409 6963966.746, 494745.647...	26081
21361	Neighbourhood / local	188.486334	LINESTRING (494760.409 6963966.746, 494731.928...	26614

(a) Relational Data



(b) Graph Data



(c) Graphical Representations in ArcGIS map

Figure 3. Utility Anchor Points and Roads, along with their Relationships

Figure 3 presents an example illustrating utility anchor points and roads, along with their relationships, in three different formats: as relational data model representation, as graph data model representation, and as visualized data on an ArcGIS map.

3.2.3 Data Encoding

Data encoding is a step to transform the data in different formats into the numerical features fed into the neural networks to predict the links between anchor point nodes. Table 1 summarizes the features used in this research, along with data encoding methods.

Table 1. Data Encoding Methods for Attributes

Node / Edge	Attribute Name	Encoding Methods
Utility Anchor Point	Location	Location Encoding
	Type	One-Hot Encoding
Road	Centroid Location	Location Encoding
	Type	One-Hot Encoding
	Length	Equal-Frequency Binning and One-Hot Encoding
	Orientation	Equal-Width Binning and One-Hot Encoding
Utility Anchor Point-Road Relationship	Relative Position of Nearest Point on Road	None
	Distance	Equal Frequency Binning and One-Hot Encoding
	Side	None

3.3 Utility Line Prediction using Graph Neural Networks

This research develops a GNN model that consists of two main components: convolutional layers, and a classifier. Initially, it adopts a multi-scale location encoder [14] that applies sinusoidal functions of varying frequencies to transform location data. The convolutional layers include the GAT (Graph Attention Network [16]) and GraphSAGE (SAmple and aggreGatE [17]) as basic building blocks. GAT incorporates an attention mechanism, assigning importance weights to neighboring nodes that are learnable within the network. It processes node features, edge indices (indicating node connections), and edge attributes as inputs and generates updated node features and attention weights as outputs. GraphSAGE is a method of sampling neighboring nodes with specific weights and aggregating these neighboring node features into the weighted target node. Both layers focus on feature aggregation at the graph nodes. The final component of the network is a binary classifier, designed to predict connections between node pairs through the multiplication of their feature vectors. The loss function used is cross-entropy loss function, commonly applied in binary classification tasks. Figure 4 presents a detailed visualization of the GNN model, including its inputs, outputs, and overall architecture.

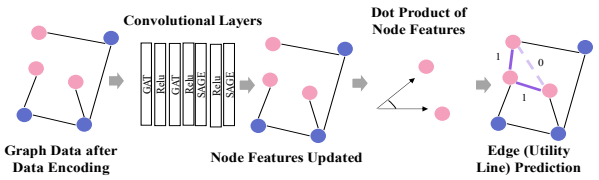


Figure 4. Architecture of the GNN model and corresponding Inputs and Outputs

3.3.1 Architecture Variants

Since there are no existing GNNs for this application, several architectural variants are discussed, as illustrated in Figure 5. ReLU layers are not drawn for simplification.

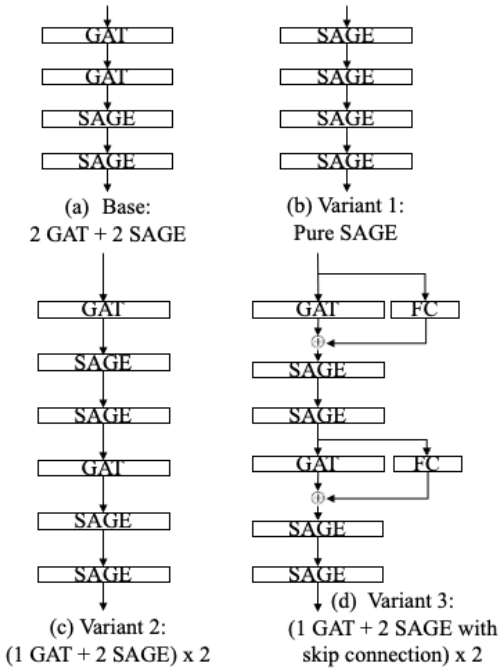


Figure 5. Architecture Variants

The base model utilizes two GAT layers, which include dropout rates to prevent overfitting. The outputs from these GAT layers, which are the updated features of the nodes, along with the indices of the edges, are fed into two GraphSAGE layers. The first variant consists solely of four GraphSAGE layers. Unlike GAT layers, GraphSAGE layers do not process edge attributes,

meaning that this model variant does not include information on road-anchor point spatial relationships beyond connectivity. The second variant examines the impact of alternating positions of GAT and SAGE layers. The third variant investigates the effective integration of edge attributes by introducing skip connections, with FC referring to fully connected layers.

4 Experimentation

4.1 Data Description and Preprocessing

This research utilizes two data sources: (1) The sewer network map provided by Urban Utilities, accessible at https://services3.arcgis.com/ocUCNI2h4moKOpKX/arcgis/rest/services/UU_Sewer_OpenData/FeatureServer. In the ArcGIS sewer network map, the manhole and pump feature layers are utilized to represent utility anchor points, and the gravity sewer main lines are used as the utility lines. (2) The road network from the Brisbane City Council, which is available at https://services2.arcgis.com/dEKgZETqwmDAh1rP/arcgis/rest/services/Roads_hierarchy_overlay_Road_hierarchy/FeatureServer. The road feature layer is employed as an example of ground facilities.

The raw data about manholes, pumps, gravity sewer main lines, and road networks were exported from ArcGIS Pro software as individual shapefiles. Subsequently, these files were processed using Python geospatial data analysis and network analysis packages. The proximity analysis between manholes and roads was conducted using the *QueryPointAndDistance* function in ArcGIS Pro Python API. This function identifies the nearest point on a polyline to a given point and calculates the distance between them. Additionally, it provides details about which side of the line the point is located on and the distance along the line, expressed as a percentage. The data was preprocessed in two steps. First, the data was cleaned by removing utility lines that lack connections with any manhole or pump points or are linked to only one point. This is because the method assumes that each utility line connects to a minimum of two anchor points. Second, roads located more than 100 meters from the manholes were filtered out, as roads not classified as "close" to the manholes do not contribute to link prediction. The statistics are summarized in Table 2.

Table 2. Data statistics before and after pre-processing

Name	Count (Before)	Count (After)
Utility Line	243,773	203,203
Utility Anchor Point	206,187	206,187
Road	41,753	32,080

4.2 Experiment Design

4.2.1 Training, Validation, and Testing Data Split

The data was divided into training, validation, and testing sets in three steps: (1) within the utility anchor point networks (excluding roads), connected components were identified, leading to a collection of subgraphs, each representing a distinct component; (2) the training, validation, and testing datasets were then randomly distributed in a 6:2:2 ratio from these subgraphs. (3) nodes representing roads were included in various datasets, determined by their connectivity to utility anchor points. Some road nodes might appear in multiple datasets if they are connected to anchor points belonging to different sets. This separation ensures that utility line edges and utility anchor point nodes from the training set do not appear in the validation or testing sets, and those from the validation set are excluded from the testing set.

This approach of using connected components for dataset division was chosen because the distribution of unknown utility lines typically concentrates in specific areas rather than being evenly spread throughout a city. Figure 6 illustrates the distribution.

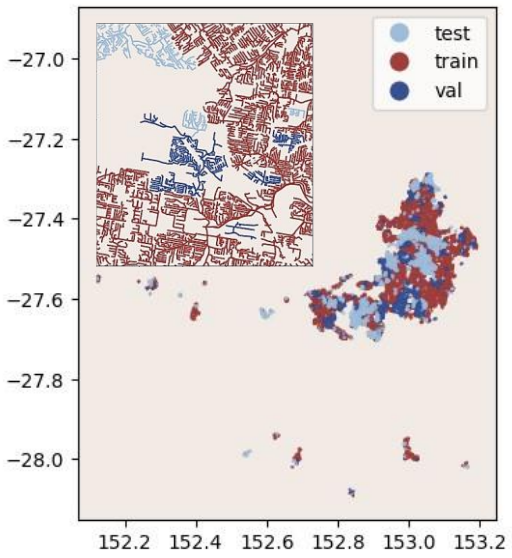


Figure 6. Training, Validation and Testing Sets

4.2.2 Evaluation Metrics

The model outputs are numerical values representing classes assigned to each edge that connects two manhole nodes: “closer” to 1 indicates the presence of a pipeline, “closer” to 0 signifies its absence. A common threshold of 0.5 is used to separate these two classes. These studies adopt the following evaluation metrics for experiments: (1) Precision. Precision is the proportion of true positive predictions, correctly predicted pipeline presence, out of

all positive predictions made. (2) Recall. Recall is the proportion of true positives, correctly predicted pipeline presence, out of all actual pipeline presences. (3) AUC (Area Under the Curve) ROC (Receiver Operating Characteristics) score. The ROC curve is a plot of the TPR (True Positive Rate or Recall) against the FPR (False Positive Rate) at various threshold settings. It is the measure of separability of two classes. (4) F1-Score. F1-Score is the harmonic mean of precision and recall. (5) Accuracy. Accuracy is the total number of correct predictions among all the cases. (6) MCC (Matthews Correlation Coefficient). MCC is a correlation coefficient between the observed and predicted classifications for imbalanced dataset. It returns a value between -1 and 1, where 1 indicates a perfect prediction, 0 means random prediction, and -1 indicates total disagreement between prediction and observations.

4.2.3 Hyperparameter Tuning

For model training, the number of epochs is determined using an early stopping approach. The maximum size of the epochs is 100, but once the validation loss does not decrease or decreases by less than 0.001 for five consecutive epochs, the training process will be stopped to prevent overfitting. Regarding optimization techniques, the Adam optimizer is used with a learning rate set at 0.001.

The model fine-tuning focuses on two hyperparameters: the size of the hidden layers and the dropout rates. This approach is chosen due to an overfitting problem observed during the experimentation process. The options for the hidden layer size are set at four specific values: 32, 64, 128, and 256. The dropout rates range from 0 to 0.6, with increments of 0.2. The hyperparameter tuning process is guided by various evaluation metrics on the validation set, and the testing data is used only for evaluating the optimal model. Due to space constraints, the detailed evaluation metrics corresponding to each model variant and hyperparameter combination are stored in the GitHub repository.

4.3 Experiment Results

The hyperparameter tuning of the model is driven by its performance on various evaluation metrics using validation data. The best-performing models on the validation set for each metric are summarized in Table 2.

For more detail, Variant 3a represents a model configuration with a hidden layer size of 32 and a dropout rate of 0; Variant 3b is configured with a hidden layer size of 128 and a dropout rate of 0; and Variant 3c features a hidden layer size of 32 with a dropout rate of 0.4. Variant 1a, on the other hand, corresponds to a model with a hidden layer size of 64, utilizing only SAGE layers.

The fine-tuned models, each selected for achieving the highest score for each evaluation metrics, are further evaluated using the testing data. The outcomes from these tests are compiled and presented in Table 3.

Overall, Variants 1 and 3 demonstrate the most robust performance. Variant 3 excels in ROC AUC scores, accuracy, and MCC metrics, indicating its superior capability in differentiating the presence and absence of pipeline connections. On the other hand, Variant 1, which focuses solely on node attributes and connectivity and overlooks edge attributes such as the manhole's location relative to the road, achieves the highest recall and F1 score. This outcome is reasonable since ignoring road-crossing pipelines leads to more conservative predictions. This conservative approach is particularly advantageous in utility line detection scenarios, where the priority is to minimize the risk of missing lines.

Table 3. Optimal Model Architectures and Hyperparameter Combinations on Validation Set

Model	Var.3a	Var. 3b	Var. 3c	Var. 1a
ROC AUC	**0.9619**	0.9616	0.9608	0.9572
F1	0.8987	0.8980	0.8959	**0.9001**
Precision	0.9137	0.9215	**0.9265**	0.8992
Recall	0.8842	0.8756	0.8672	**0.9010**
Accuracy	0.9004	**0.9005**	0.8992	0.9000
MCC	0.8011	**0.8020**	0.8001	0.8000

Table 4. Testing Results on the Tunned Models

Model	Var.3a	Var. 3b	Var. 3c	Var. 1a
ROC AUC	0.9520	**0.9524**	0.9488	0.9479
F1	0.8855	0.8849	0.8790	**0.8868**
Precision	0.8783	0.8884	**0.8890**	0.8687
Recall	0.8927	0.8815	0.8692	**0.9057**
Accuracy	0.8845	**0.8854**	0.8803	0.8844
MCC	0.7692	**0.7708**	0.7608	0.7695

5 Conclusion and Discussion

This research presents an effective method for completing utility networks. The approach includes three steps: (1) build a relational data model to arrange the data regarding utility anchor points, lines, ground facilities, and their spatial relationships; (2) convert all records in the relational data model to graphs, with anchor points and facilities as nodes, and utility lines and their relationships as edges. (3) develop a GNN model to predict utility lines. The experimental results demonstrate good performance, achieving a 95.2% ROC AUC score in inferring sewer lines between manholes.

This novel approach offers advantages for utility owners and excavation contractors, providing a framework to deduce missing connections within utility networks.

However, a limitation of the model is its lack of explainability, which impacts user trust. Furthermore, applying the model directly to varied datasets presents challenges due to the necessity for: (1) aligning context features with standardized utility network criteria, and (2) considering diverse practices that vary by time and geography. Ensuring model adaptability to different utility networks requires accurate, complete, and region-specific utility network training data. Future research will focus on assessing the impact of data quality on model performance. Additionally, expanding the model to include more spatial contexts, such as buildings and legal boundaries, could further improve its utility and accuracy in real-world applications. Lastly, considering potential consequences of false alerts and missed detections in utility strike prevention and flexibilities in pipeline network design, presenting the likelihood with uncertainty could further improve decision making.

6 Data and Code Availability

The code, data, and supplemental materials are available in the GitHub repository: https://github.com/Yuxi0048/PipeNetworkCompletion.

References

[1] Damage Root Causes Remain Consistent. https://dirt.commongroundalliance.com/2022-DIRT-Report/Damage-Root-Causes-Remain-Consistent#mainContentAnchor, Accessed: 24/12/2023

[2] Afshar, M. H., Partially Constrained Ant Colony Optimization Algorithm for the Solution of Constrained Optimization Problems: Application to Storm Water Network Design, *Advances in Water Resources*, Vol. 30, No. 4, 2007, pp. 954–965. https://doi.org/10.1016/j.advwatres.2006.08.004

[3] Izquierdo, J., Montalvo, I., Pérez, R., and Fuertes, V. S., Design Optimization of Wastewater Collection Networks by PSO, *Computers & Mathematics with Applications*, Vol. 56, No. 3, 2008, https://doi.org/10.1016/j.camwa.2008.02.007

[4] Bailly, J. S., Levavasseur, F., and Lagacherie, P., A Spatial Stochastic Algorithm to Reconstruct Artificial Drainage Networks from Incomplete Network Delineations, *International Journal of Applied Earth Observation and Geoinformation*, Vol. 13, No. 6, 2011, https://doi.org/10.1016/j.jag.2011.06.001

[5] F. Bazlamaçcı, C., and S. Hindi, K., Minimum-Weight Spanning Tree Algorithms A Survey and Empirical Study, *Computers & Operations Research*, Vol. 28, No. 8, 2001, https://doi.org/10.1016/S0305-0548(00)00007-1

[6] Navin, P. K., and Mathur, Y. P., Layout and Component Size Optimization of Sewer Network Using Spanning Tree and Modified PSO Algorithm, *Water Resources Management*, Vol. 30, No. 10, 2016, https://doi.org/10.1007/s11269-016-1378-7

[7] Chahinian, N., Delenne, C., Commandré, B., Derras, M., Deruelle, L., and Bailly, J.S., Automatic Mapping of Urban Wastewater Networks Based on Manhole Cover Locations, *Computers, Environment and Urban Systems*, 2019.

[8] LeCun, Y., Bengio, Y., and Hinton, G., Deep Learning, *Nature*, Vol. 521, No. 7553, 2015, https://doi.org/10.1038/nature14539

[9] Alzubaidi, L., Zhang, J., Humaidi, A. J., Al-Dujaili, A., Duan, Y., Al-Shamma, O., Santamaría, J., Fadhel, M. A., Al-Amidie, M., and Farhan, L., Review of Deep Learning: Concepts, CNN Architectures, Challenges, Applications, Future Directions, *Journal of Big Data*, Vol. 8, No. 1, 2021, https://doi.org/10.1186/s40537-021-00444-8

[10] Zhang, M., and Chen, Y., Link Prediction Based on Graph Neural Networks, Vol. 31, 2018. *Advances in neural information processing systems, 31*.

[11] Hamilton, W. L., Ying, R., and Leskovec, J., Representation Learning on Graphs: Methods and Applications, 2017. *arXiv preprint arXiv:1709.05584*.

[12] Goodfellow, I., Bengio, Y., and Courville, A., *Deep Learning*, MIT Press, 2016.

[13] Wang, X., Ma, Y., Wang, Y., Jin, W., Wang, X., Tang, J., Jia, C., and Yu, J., Traffic Flow Prediction via Spatial Temporal Graph Neural Network, New York, NY, USA, 2020. https://doi.org/10.1145/3366423.3380186

[14] Wang, S., Li, Y., Zhang, J., Meng, Q., Meng, L., and Gao, F., PM2.5-GNN: A Domain Knowledge Enhanced Graph Neural Network for PM2.5 Forecasting, New York, NY, USA, 2020. https://doi.org/10.1145/3397536.3422208

[15] Mai, G., Janowicz, K., Yan, B., Zhu, R., Cai, L., and Lao, N., Multi-scale representation learning for spatial feature distributions using grid cells, 2020. *ICLR 2020.*

[16] Veličković, P., Cucurull, G., Casanova, A., Romero, A., Liò, P., and Bengio, Y., Graph Attention Networks, 2018. *ICLR 2018.*

[17] Hamilton, W. L., Ying, R., and Leskovec, J., Inductive Representation Learning on Large Graphs, 2017. *Advances in neural information processing systems, 30*.