Vector generalization of the drainage network

Thalles Oliveira Barros de Aquino[1], Edilson Bias[2], Maurício Carvalho Mathias de Paulo[3], Raul Queiroz Feitosa[4], Felipe Ferrari[5]

[1] Institute of Geoscience, University of Brasília, Brasília, Brazil - thalles.oliveira@icloud.com
[2] Institute of Geoscience, University of Brasília, Brasília, Brazil - edbias@unb.br
[3] Cartographic Engineering Department, Military Institute of Engineering, Rio de Janeiro, Brazil - mauricio.paulo@eb.mil.br
[4] Dept. of Electrical Engineering, Pontifical Catholic University, Rio de Janeiro, Brazil - raul@ele.puc-rio.br
[5] Cartographic Engineering Department, Military Institute of Engineering, Rio de Janeiro, Brazil - felipeferrari@eb.mil.br

Keywords: River Network Generalization, Graph Convolutional Networks (GCN), GraphSAGE, Geo-AI (Geospatial Artificial Intelligence)

Abstract

Automated map generalization at regional scales remains a bottleneck in national mapping agencies. Generalization rules for hydrographic networks are complex and difficult to quantify using direct indicators. To address generalization challenges in drainage and examine how non-visual information can improve feature selection, this study evaluates and adapts an automated Graph Convolutional Network (GCN) approach, focusing on the GraphSAGE model proposed by (Wang and Qian, 2023). The hydrographic network is represented as a graph, with nodes corresponding to drainage segments and edges representing their connections. Segment classification, as retained or discarded, is based on semantic, geometric, morphological, topological, and constraint-related attributes. The method was applied to drainage generalization using four attribute sets derived from the Brazilian Technical Specifications of the Geospatial Vector Data Structure (ET-EDGV). Three geometric attributes, length, sinuosity, and polygon containment, were tested together with one non-geometric attribute: fluvial regime. Using only geometric attributes, the best model achieved 94.52% training accuracy, 94.70% validation accuracy, 93.91% Recall, and a 95.30% F1 Score. When the fluvial regime was included, performance increased to 99.98% test accuracy, 99.98% validation accuracy, 99.98% Recall, and a 99.98% F1 Score. The generalized network was validated against a 1:100,000 reference drainage dataset manually generalized from the same 1:25,000 input, yielding 67.16% positional accuracy for the model trained only with geometric attributes. Results indicate that GraphSAGE is suitable for selecting drainage segments from ET-EDGV attributes, especially for scale transitions from 1:25,000 to 1:100,000.

1. Introduction

Automatic map generalization remains a fundamental challenge in cartography, particularly when dealing with complex geographic features such as hydrographic networks. These features exhibit rich semantic, geometric, and topological characteristics that are often interdependent and context sensitive, making it difficult to encode their generalization rules using fixed algorithms or visual criteria alone. Manual methods, while effective, are time-consuming, prone to subjectivity, and lack scalability.

Within this context, frameworks such as the Technical Specifications of the Geospatial Vector Data Structure (ET-EDGV) play a critical role in standardizing the representation of spatial data, particularly in Brazil. ET-EDGV enables consistent, interoperable, and high-quality vector data modeling, which is essential to ensure the integrity and usability of the generalization output (CONCAR, 2010). Its integration within the National Spatial Data Infrastructure (INDE), Brazil’s national spatial data infrastructure, further reinforces the importance of standard-based coordinated data governance (CINDE, 2010).

Aligned with INDE’s directives, the Brazilian Army’s solution for presenting generalized data across different web clients is the Geodata Loader for MapServer (GDLMS). This system employs `.map` file templates and builds on previous indexing strategies to manage overlaps. GDLMS automates visibility decisions, hiding outdated or superseded geometries based on spatial extent and metadata.

However, despite this technical sophistication, much of the system still relies on manual input—particularly in populating metadata fields in accordance with the ET-EDGV standard. This approach relies on cartographic generalization rules or optimization techniques, such as simplification, selection, and displacement algorithms, which are manually encoded and adapted to specific map features (Silva and Centeno, 2023). Although these methods are interpretable and can produce high-quality results, they struggle with scalability and adaptability, especially when applied to complex and diverse datasets (Castro and Sluter, 2014).

A more recent and promising direction involves the use of Graph Neural Networks (GNNs), particularly Graph Convolutional Networks (GCNs), which extend deep learning techniques to graph-structured data. In this context, geographic features such as river networks or road systems are modeled as graphs, where nodes represent individual segments and edges encode their topological relationships. GCNs, and specifically the GraphSAGE model, offer a way to learn feature representations by aggregating information from a node’s neighbors, enabling the model to incorporate both local and contextual information (Hamilton et al., 2018). These spatial patterns and relationships are difficult to capture with traditional rule-based approaches (Xiao et al., 2025). Approaches like this have been proven effective for tasks such as segment classification and network simplification, enabling more intelligent, structure-aware generalization (Hamilton et al., 2018).

In response to these challenges, we evaluated the performance of the GraphSAGE network proposed in (Wang and Qian,

2023), adapting it to the ET-EDGV attribute schema to generalize hydrographic vector data from 1:25,000 to 1:100,000 scale. This approach demonstrates the potential of a novel methodology for automatic cartographic generalization, aligned with Brazilian geospatial data standards.

The objective of this work is to evaluate and adapt the GraphSAGE network proposed by (Wang and Qian, 2023) to the ET-EDGV schema for the automatic generalization of hydrographic vector data from the 1:25,000 to the 1:100,000 scale. In addition, the study investigates the feasibility of automating generalization in scenarios where categorical semantic information is absent or incomplete, assessing whether geometric, morphological, topological, and constraint-related features alone can support effective generalization.

This paper is organized as follows: Section 2 presents a brief overview of a SQL-based method that employs manually defined attributes to perform cartographic generalization, which served as a reference for the proposed neural network, along with a concise explanation of the GraphSAGE framework. Section 3 details the data construction process and the extraction of geometric features. Section 4 describes the model architecture and the training procedure. Section 5 presents the experimental results, Section 6 concludes and outlines future studies, and Section 7 acknowledges the support provided by the institutions involved in this research.

2. Related work

2.1 GDLMS - Geographic Data Loader for MapServer

GDLMS is a robust, EDGV-compliant system designed to manage vector data visibility in multiscale WMS services. The system integrates metadata, SQL-based rule logic, spatial indexing, and MapServer templates to automate the otherwise manual and error-prone tasks of importing, indexing, and preparing geospatial vector data for multiscale map services. The main goal is to determine the minimum usable scale at which each feature should be visible, ensuring consistent cartographic representation and performance across various zoom levels.

To support this goal, the multiscale index logic component calculates which features should be visible at each scale by generating an index, typically as a PostGIS table or view. It identifies the most detailed and recent data to decide which set of features will remain visible on a less detailed map, following the hypothesis that hydrographic objects to be shown at the target scale can be selected from the hydrographic objects shown at the source scale (Gökgöz et al., 2025).

This visibility attribute enables MapServer to dynamically select the most appropriate features during WMS tile rendering, ensuring accurate and efficient multiscale visualization.

Complementing the index logic, scale management is controlled by the `minusablelevel` attribute, which indicates the lowest map scale at which a feature should appear. If the current zoom level is broader than this scale, the feature is excluded from rendering. For features that must always appear, `10,0000,000` is used. For the 1:25,000 scale, it uses a `minusablelevel` of `50,000`, and for the 1:100,000 scale, a `minusablelevel` of `500,000`.

These values are defined according to ET-EDGV specifications and are manually entered by trained operators to ensure semantic fidelity. Based on these values, MapServer resolves zoom levels to actual scale denominators. This dependence on human intervention is one motivation for seeking more automated generalization methods, such as those explored in this work.

2.2 GraphSAGE

Traditional node embedding techniques, such as matrix factorization or random-walk-based models, are transductive in nature, meaning they require the entire graph structure during training and cannot generalize to unseen nodes. This limitation poses a challenge in dynamic settings such as social networks and recommendation systems. To address this, GraphSAGE (SAmple and aggreGatE) was introduced by Hamilton et. al. as a method for inductive representation learning on large graphs. It generates embeddings using a trainable aggregation function, enabling the model to infer embeddings for nodes unseen during training (Hamilton et al., 2018).

GraphSAGE samples a fixed-size neighborhood for each node, enabling scalability even in graphs with high-degree nodes. Starting from initial node features, such as text attributes or numerical data, the model uses a layer-wise architecture. In each layer, a node aggregates information from its neighbors using a parameterized function, such as mean, max pooling, or an LSTM-based aggregator. This aggregated vector is then concatenated with the node’s current representation and transformed by a learnable function. The stacking of these layers captures multi-hop context, leading to powerful embeddings that outperformed traditional GCNs and static methods in several benchmarks (Hamilton et al., 2018).

As an example of its use in mapping, recent studies on road network generalization demonstrate that graph neural networks, including the GraphSAGE model, can integrate spatial-domain knowledge with learned representations to improve the automation and structural preservation of generalized maps (Xiao et al., 2024).

3. Materials and methods

3.1 Modeling a river network as a graph

Modeling a river network as a graph involves representing the hydrological system using nodes and edges, where each node represents an individual drainage segment (topological edge), and the edges represent the topological connectivity between these segments (Wang and Qian, 2023). This abstraction enables the use of graph theory and machine learning techniques to analyze and predict hydrological behavior. For example, each node can encapsulate attributes such as elevation, flow rate, and water quality, while edges can store segment lengths and flow directions (Yan et al., 2023).

The river network is inherently a directed graph as water flows from sources to sinks. It is often modeled as a tree-like structure, with tributaries merging into the main rivers. This Directed Acyclic Graph (DAG) structure ensures there are no loops, mirroring real-world hydrology. In Graph Neural Network (GNN) applications, directionality helps capture upstream and downstream dependencies, which are crucial for tasks such as flood prediction or pollution tracking (Wang and Qian, 2023).

Figure 1 illustrates the basic structure of the river network graph. It shows nodes representing source points, confluences,

and mouths, connected by directed edges that indicate water flow, making it suitable for graph-based processing and analysis.

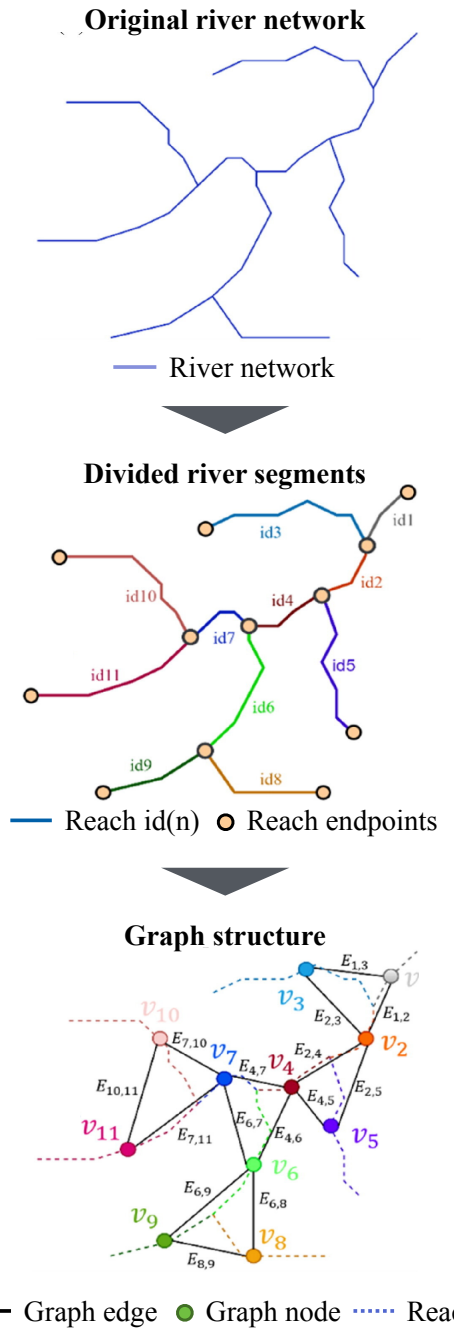

Figure 1. Sample of river network in a graph context (Wang and Qian, 2023).

3.2 Dataset

The data used in this study were obtained from the Brazilian Army’s Geographic Database (BDGEx). It is modeled according to ET-EDGV parameters. As this work focuses on drainage generalization, we chose 96 charts of an area mapped at a 1:25,000 scale from the state of Santa Catarina, Brazil, as shown in Figure 2.

These 1:25,000-scale charts were used as input data from

Figure 2. Area of Interest.

December 2016, and their editions were completed in 2017 and 2018.

The drainage network data was imported into a PostgreSQL database with the PostGIS extension. To extract the key geometries required for constructing the graph structure, we created a topology schema to generate a table of topology nodes containing the start and end points of each segment, as well as their intersections, and a table of topology edges, where each edge corresponds to a drainage segment. Those topology nodes and the graph structure, yet to be elucidated. They are the elements used to create the graph nodes and graph edges.

In total, 199,860 drainage segments were identified. From the GDLMS generalized data on a scale of 1:100,000, 117.580 segments were retained as visible.

We also used drainage vector data from six topographic charts covering the same area of interest, originally produced at the 1:100,000 scale, as a reference dataset to evaluate the positional accuracy of the generalized network. These charts were produced in 2022 through a manual generalization process, using the topographic charts from the previous mapping campaign (2016-2018) as input, originally compiled at the 1:25,000 scale for the same region.

3.3 Preprocessing

Those tables in the topology schema were used to create the `graph_nodes` and `graph_edges` tables.

Each topological edge contains the `id` and the geometry of its corresponding drainage segment. These are assigned as the `id` and geometry attributes of the `graph_nodes`. Furthermore, a binary column `y` was added to the `graph_nodes`, derived from the multiscale generalized dataset (GDLMS), indicating which segments on the scale 1:25,000 should be omitted in the generalization to the scale 1:100,000.

From this `graph_nodes` table, we generate 4 other tables by adding the attributes length, sinuosity, contained inside polygon, and fluvial regime in new columns, in this order. For example, the third table, called Inside_polygon, has its first attribute column populated with the length, the second with the

This contribution has been peer-reviewed. The double-blind peer-review was conducted on the basis of the full paper.
https://doi.org/10.5194/isprs-annals-XI-4-2026-77-2026 | © Author(s) 2026. CC BY 4.0 License.

Table 1. Tables used to train the model

Graph nodes table name	Set of attributes
Length	Length
Sinuosity	Length, sinuosity
Inside_polygon	Length, sinuosity, contained in polygon
Regime	Length, sinuosity, contained in polygon, fluvial regime

sinuosity, and the third with the ’contained inside a polygon’ attribute. Table 1 shows the tables created:

The Inside Polygon attribute is derived from a spatial overlay analysis of hydrographic features defined in the ET-EDGV specification. It is cartographically relevant because drainage segments contained within larger water bodies (hydrographic polygons) often require differentiated treatment during generalization, such as preferential retention or contextual simplification, to maintain visual coherence and spatial information. If the segment is enclosed within such a polygon, the attribute is assigned the value 1; otherwise, it receives a value of 0.

Also present as an attribute in the ET-EDGV, the fluvial regime is categorical, human-defined, and relies on expert interpretation of river flow characteristics rather than direct measurement. As such, its classification is inherently subjective and may be influenced by regional practices or individual judgment.

The fluvial regime attribute contains four distinct categories, each encoded as an integer value. Table 2 presents the mapping between these categories and their corresponding numerical representations.

Table 2. Numerical representations of the fluvial regime

fluvial regime	Value
Dry	0
Temporary	1
Permanent	2
Permanent with high variation	3

The length L of a river segment represented by a `LINESTRING` geometry can be calculated using the spatial function `ST_Length`.

We used the Index of Total Sinuosity (Mueller, 1968), studied by Mueller in 1968 and used by Stanislawski at 2023 (Stanislawski et al., 2023). Here is the formula:

$$\kappa = \frac{L}{D} \tag{1}$$

Where:

- L is the length of the curve (e.g., river segment); and
- D is the Euclidean distance between the start and end points of the curve.

This formula results in a dimensionless ratio, where:

- $\kappa = 1$ implies a perfectly straight segment.
- Larger κ values indicate more sinuosity.

On the other hand, the `graph_edges` table consists of two columns: the first column, `u`, contains the `id` of a `graph_nodes` element, and the second column, `v`, contains the `id` of its adjacent `graph_nodes` element.

This table was constructed by performing a self-join on the topological edges, where the endpoint of one segment matches the start point of another.

4. Experimental setup

4.1 Model

The algorithm, implemented in Python with PyTorch Geometric, aims to classify each node in a drainage network graph. It employs layers called `SAGEConv`, which implement neighborhood aggregation: each node updates its features by aggregating its neighbors’ features.

These layers perform neighborhood aggregation. Specifically, the model uses max pooling as its aggregation function, selecting the maximum feature value from each node’s neighbors to form the new representation.

The architecture includes three sequential GraphSAGE layers. The first layer, `conv1` , transforms the input features into a hidden representation. The `conv2` layer refines this representation further, and the third layer, `conv3` , projects it into an output space corresponding to the target classes. These convolutional layers are followed by a single fully connected (Linear) layer, which provides additional feature transformations and helps stabilize learning. Between each graph convolution layer, a ReLU activation introduces non-linearity, enabling the model to capture complex patterns. The final output is passed through a log-softmax function, which converts the raw logits into log-probabilities suitable for classification, particularly for multi-class node classification in the river network graph.

4.2 Training and evaluation

The model was trained four times; in each case, it received the `graph_edges` table and one of the four tables with different node attribute values as input. It was necessary to adjust the tensor’s dimensionality parameters for each training run.

The training process was guided by a masking strategy, in which the dataset was split into training (64%), validation (16%), and test (20%) subsets using random sampling with a fixed seed (98). Only nodes marked `True` on the `train_mask` are used to compute the loss and update the model’s parameters. The `val_mask` and `test_mask` identify nodes held during training and are used exclusively for validation and final evaluation, respectively. This separation ensures that the model learns from a distinct subset of labeled nodes, while generalization is assessed on unseen data.

Model optimization was handled by the Adam optimizer (learning rate 0.005), a widely used gradient-based method known for its efficiency and adaptability. In this case, it also includes `L2 regularization`, controlled by the `weight_decay` parameter of 5×10^{-4}, to prevent overfitting by penalizing overly complex models. This training configuration ensures a robust and generalizable model for node classification in river networks.

To improve training efficiency and prevent overfitting, an early stopping mechanism with a patience of 100 epochs was incorporated into the code provided by (Wang and Qian, 2023).

This contribution has been peer-reviewed. The double-blind peer-review was conducted on the basis of the full paper.
https://doi.org/10.5194/isprs-annals-XI-4-2026-77-2026 | © Author(s) 2026. CC BY 4.0 License.

This mechanism halted training once the validation performance stopped improving for 100 consecutive epochs. As shown in Table 3, the training successfully triggered early stopping after a certain number of epochs, indicating convergence before reaching the maximum training limit. The early stopping criterion was defined as: validation accuracy must exceed the best recorded validation accuracy.

Table 3. Number of epochs for each training

Table name	Epochs
Length	114
Sinuosity	188
Inside Polygon	182
Regime	149

We represented all the features that should appear at 1:100,000 with a Boolean label, as in GDLMS generalization. For each of the four training runs, we exported the loss chart and the predictions **y** to be added to the drainage table for visualization in GIS software.

4.3 Network position accuracy

Once the 1:25,000-scale data was generalized to the 1:100,000 scale using GDLMS or GraphSAGE, it was compared with the drainage vector lines on the 1:100,000 topographic charts. Positional accuracy for network definition was recently elucidated by Han et al. in 2024 in their study of road networks, defined as the degree of geographic location deviation between the elements in the dataset to be evaluated and those in the reference dataset (Han et al., 2025). The network positional accuracy can be calculated using the following formula:

$$Acc_{\text{gen}} = \frac{L_{\text{gen}} \cap Buffer_{\text{ref}}}{L_{\text{gen}}} \times 100\% \quad (2)$$

Where:

- Acc_{gen} is the dimensionless network accuracy.
- L_{gen} is the length of the generalized drainage network; and
- $Buffer_{\text{ref}}$ is the buffer of the reference data, originally acquired on a scale of 1:100,000.

We used a buffer radius of 0.2 millimeters, using the plotted chart on paper as a reference. It is a well-established cartographic threshold for the minimum spacing between disjoint map objects in a chart (graphicism error (Jordão et al., 2022)), making it a suitable criterion to evaluate the accuracy and visual separability of the network on the target scale (Harrie et al., 2015). At a 1:100,000 scale, this represents a distance of 20 meters in real life.

For these measures, the features were reprojected to EPSG:31982 before the buffer and intersection operations.

5. Results and discussion

Some results were generated from the 96 1:25,000 charts generalized to a 1:100,000 scale.

Figure 3 shows the training loss chart for each model.

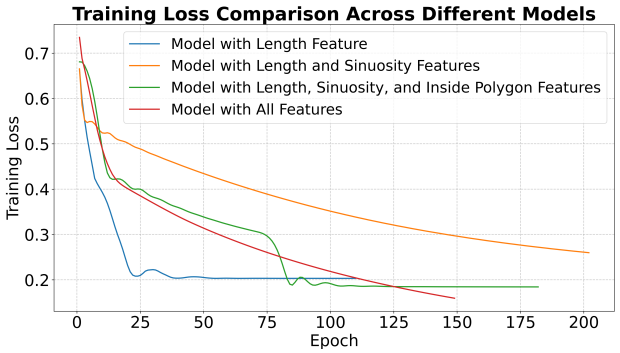

Figure 3. Training Loss Comparison Across Different Trainings.

The model incorporating all attributes, including geometric features and the categorical fluvial regime, consistently achieved the lowest training loss, indicating more efficient convergence and superior learning capacity. Notably, the model was trained only on Length and Sinuosity features. The addition of the Inside Polygon feature led to a noticeable improvement, particularly after epoch 75, when a sharp drop in loss indicated a better model fit. Meanwhile, the model trained only on the Length feature also outperformed the Length-and-Sinuosity configuration, suggesting that including Sinuosity alone may add complexity without significantly improving performance. In general, the results confirm that the combination of geometric and semantic information leads to more effective training, with the fluvial regime attribute playing a key role in accelerating convergence and minimizing training error.

Table 4 represents the value of some statistics.

Table 4. Model performance across different feature sets

Metric (Best values)	Length	Sinuosity	Inside Polygon	Regime
Train Loss	0.20	0.26	0.18	0.16
Val Loss	0.20	0.26	0.18	0.16
Train Acc (%)	93.87	93.88	94.52	99.99
Val Acc (%)	93.94	93.95	94.70	99.98
Test Acc (%)	93.93	93.93	94.56	99.98
Precision (%)	94.66	94.66	96.73	99.98
Recall (%)	95.00	95.00	93.91	99.98
F1 Score (%)	94.83	94.83	95.30	99.98
AUC (%)	93.70	93.70	94.70	99.98

The performance results across different feature sets, as presented in Table 4, reveal that the model trained on all attributes, including the fluvial regime, significantly outperforms the others. It achieves the lowest training and validation losses (0.160), coupled with the highest accuracy in the training (99.99%), validation (99.98%), and testing (99.98%) phases. Furthermore, its precision (99.98%), recall (99.98%), F1 score (99.98%), and area under the curve (AUC) (99.98%) indicate exceptional classification consistency and minimal misclassification.

These metrics collectively suggest that the regime feature encodes highly discriminative information, enabling the model to generalize effectively to unseen data. Compared to each other, the trainings using the Length, Sinuosity, and Inside Polygon feature sets perform similarly, with marginally lower metrics. Among them, when the model was trained using the Inside Polygon table (with length, sinuosity, and inside polygon features), it performed slightly better, showing the lowest validation loss (0.184); the highest train (94.51%), test (94.56%),

This contribution has been peer-reviewed. The double-blind peer-review was conducted on the basis of the full paper.
https://doi.org/10.5194/isprs-annals-XI-4-2026-77-2026

and validation (94.70%) accuracy; and also the highest F1 score (94.0%) within this group. However, recall was better when the model was trained with the length attribute alone (95,00%). Although these features contribute positively to the classification task, their effectiveness is lower than that of a model trained on the fluvial regime attribute.

Figure 4 presents a rare instance of divergence between the generalizations produced by the GraphSAGE model trained with all attributes, including fluvial regime, and those produced by GDLMS. Since the test selection was random, Figure 4 shows a sample of the predictions. In this example, both resulting vectors are overlaid within a GIS environment (QGIS-LTR 3.40.5-Bratislava). The information in the maps is at a 1:100,000 scale. Although there is a zoom in the second map, the data remain the same as in the first. The red segment highlights a false negative example. It’s a feature that GraphSAGE chose to omit at a smaller scale, whereas GDLMS retained it, resulting in a visible difference in the level of simplification applied during model training.

Following the same logic and structure as Figure 4, Figure 5 shows examples of divergence between the GraphSAGE model trained only on geometric attributes (length, sinuosity, and whether it is contained by a polygon) and GDLMS generalizations. Once again, we have both vector layers, from GraphSAGE and GDLMS, in the same area shown in Figure 4, stacked in the same GIS software. The red segments represent the GraphSAGE segments it decided to omit, while GDLMS chose to keep them visible.

In addition, GDLMS generalization itself and the results were compared with the originally acquired data in a 1:100,000 scale from the systematic cartographic reference charts (human-made generalization). Table 5 presents the accuracy of the drainage network (Han et al., 2025) for the generalizations of the GDLMS and each of the four model results (length, sinuosity, inside polygon, and regime models), along with their total feature length and the length of the intersections to the reference data buffer of 20 meters.

Table 5. Drainage Network Accuracy

Model (Generalization)	**Total length** (km)	**Inside buffer** (km)	**Network accuracy**
GDLMS	30,326	21,111	69.61%
Length	29,283	19,566	66.82%
Sinuosity	29,287	19,566	66.81%
Inside Polygon	28,986	19,468	67.16%
Regime	30,223	19,763	65.18%

These results in Table 5 show that the GDLMS logic itself achieves only 69.61% agreement with the human-generalized 1:100,000 drainage network, even in a well-attribute area selected as the area of interest. Thus, the fact that the length model reproduced 93.5% of the GDLMS selection logic should be understood as strong adherence to an automatic procedure that, in turn, is limited in its capacity for human generalization.

In light of these results, an important next step is to improve the selection of drainage segments for training data. A promising approach would be to refine the selection of GDLMS drainage by considering whether the start and end points of each segment intersect the reference buffer, thereby favoring segments that more closely correspond to the human-generalized network. Such a strategy could provide a more consistent training set, allowing the length model to learn a selection pattern closer to

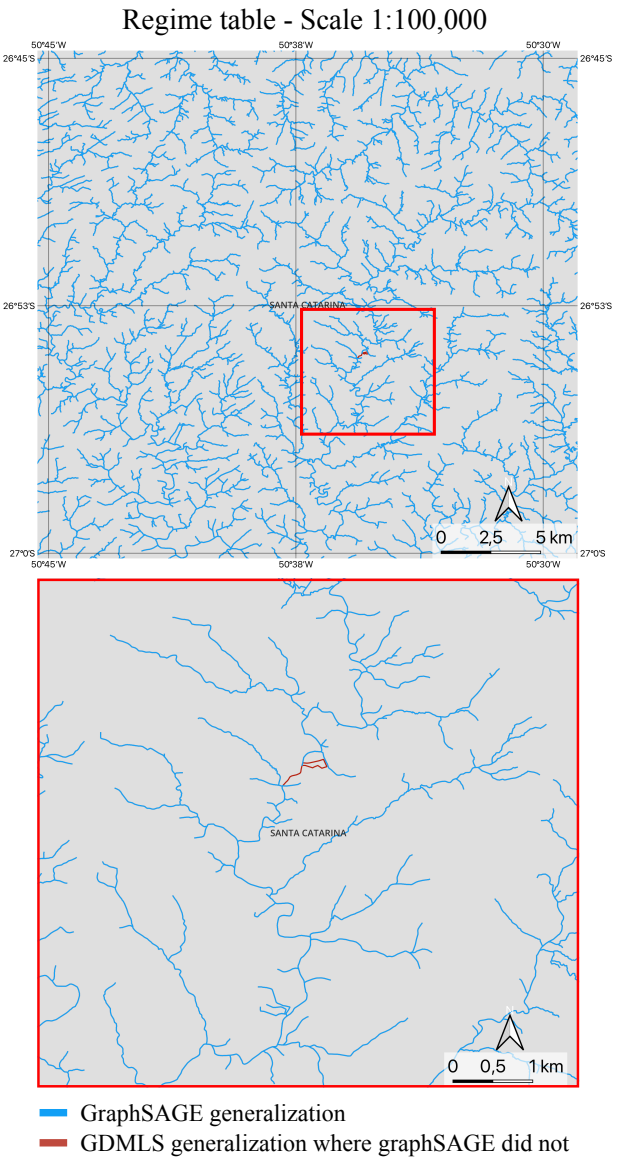

Figure 4. All attributes (including fluvial regime) model predictions compared to the GDLMS generalization. In red a example of a false negative.

human cartographic practice and making it more reliable for use in areas lacking ET-EDGV attribution, where length may be the only consistently available attribute.

For further analysis, the complete notebook is available here.

6. Conclusion

In this study, a graph convolutional network model was trained four times, each time using a different combination of attributes. Three of these trainings were fed exclusively with geometric features such as length, sinuosity, and whether the segment is contained within a hydrographic polygon. The fourth training included, in addition to these, the categorical attribute of the fluvial regime. The aim was to assess the impact of each feature set on the performance of the GraphSAGE model, proposed as a deep learning approach to the cartographic generalization of drainage segments. The results were compared

This contribution has been peer-reviewed. The double-blind peer-review was conducted on the basis of the full paper.
https://doi.org/10.5194/isprs-annals-XI-4-2026-77-2026

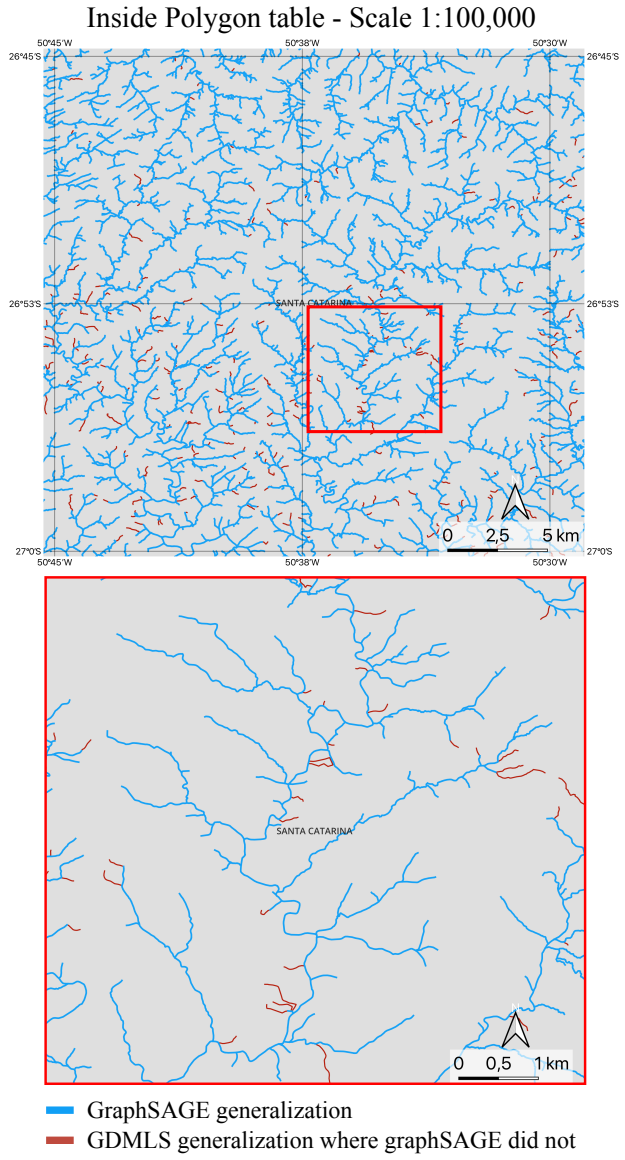

Figure 5. Only geometric attributes (length, sinuosity, contained in a polygon) model predictions compared to the GDLMS generalization. In red, an example of a false negative.

to those obtained using GDLMS, the tool currently employed by the Brazilian Army’s Geospatial Database, with particular focus on scale transitions from 1:25,000 to 1:100,000.

The application of the proposed generalization method, using only geometric attributes, to the drainage network data resulted in a trained model whose predictions achieved training accuracy of 94.52%, validation accuracy of 94.70% and testing accuracy of 94.56%. These results are consistent with the approximately 95% accuracy reported in the previous study by (Wang and Qian, 2023), demonstrating the model’s robustness and reliability.

Furthermore, this approach provides a consistent alternative for generalization tasks when categorical, non-geometric data are unavailable.

Analysis of the different attribute combinations tested revealed that, even with a reduced set of geometric features, such as length and sinuosity, the model maintained performance above 93%. This suggests that such geometric attributes are highly informative for cartographic generalization tasks involving linear networks. The high level of accuracy achieved indicates that graph neural network models can effectively capture complex structural patterns solely from geometric information.

One of the main insights from this study is that, although including the flow regime attribute improves model performance, it may not be strictly necessary. Depending on the project’s budget or the map’s objective, an accuracy of around 94% might be enough.

This finding opens several avenues for future research. One possible direction is to investigate the implications of the approximately 6% reduction in drainage-segment generalization performance when the flow-regime attribute is unavailable.

A second research direction may explore alternative approaches to inferring flow regimes from remote sensing data, such as Digital Elevation Models (DEMs), time series of optical or radar imagery, and watershed classification techniques. These methods could eliminate the reliance on manually labeled attributes and potentially push the model’s accuracy even closer to 100%.

In addition, another future work could investigate spatially disjoint data partitioning strategies, such as splits based on subbasins or map sheets, to provide a more rigorous assessment of generalization capability.

7. Acknowledgments

The authors would like to thank the Brazilian Army for its institutional support and for providing the data used in this study.

References

Castro, M. C. d., Sluter, C. R., 2014. Propuesta de metodología para la generalización cartográfica para el mapeo topográfico de áreas urbanas. *Revista Cartográfica*, 9–26. https://revistasipgh.org/index.php/rcar/article/view/461.

CINDE, C. d. P. d. I., 2010. Plano de Ação para Implantação da Infraestrutura Nacional de Dados Espaciais. Ministério do Planejamento, Orçamento e Gestão.

CONCAR, C. N. d. C., 2010. *Especificação Técnica para a Estruturação de Dados Geoespaciais Vetoriais de Defesa da Força Terrestre (ET-EDGV)*. Ministério do Planejamento, Orçamento e Gestão.

Gökgöz, T., Hacar, M., Şengün, Y. S., Simav, , Çobankaya, O. N., 2025. Uncertainty in Generalization: A Comparative Analysis of Selection/Elimination by Algorithms and Cartographers. *Transactions in GIS*, 29(7), e70143. https://onlinelibrary.wiley.com/doi/abs/10.1111/tgis.70143.

Hamilton, W. L., Ying, R., Leskovec, J., 2018. Inductive Representation Learning on Large Graphs. NIPS 2017. arXiv:1706.02216 [cs] version: 4 http://arxiv.org/abs/1706.02216.

Han, C., Lu, B., Zheng, J., Yu, D., Zheng, S., 2025. Research on multiscale OpenStreetMap in China: data quality assessment with EWM-TOPSIS and GDP modeling. *Geo-spatial Information Science*, 28(3), 1316–1340. https://doi.org/10.1080/10095020.2024.2356238.

Harrie, L., Stigmar, H., Djordjevic, M., 2015. Analytical Estimation of Map Readability. *ISPRS International Journal of Geo-Information*, 4(2), 418–446. https://www.mdpi.com/2220-9964/4/2/418.

Jordão, N., Pinhal, A., Gonçalves, J. A., 2022. Improved Methodologies for the Revision of a Traditional Topographic Map Series. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XLIII-B4-2022, 357–362. https://isprs-archives.copernicus.org/articles/XLIII-B4-2022/357/2022/isprs-archives-XLIII-B4-2022-357-2022.html.

Mueller, J. E., 1968. An Introduction to the Hydraulic and Topographic Sinuosity. *Annals of the Association of American Geographers*, 58(2), 371–385. https://onlinelibrary.wiley.com/doi/abs/10.1111/j.1467-8306.1968.tb00650.x.

Silva, J. R. D., Centeno, J. A. S., 2023. Representação de Cartas Topográficas em Dispositivos Móveis de Realidade Aumentada Aplicando Conceito de Generalização Cartográfica. *Rev. Bras. Cartogr.*, 75. https://seer.ufu.br/index.php/revistabrasileiracartografia/article/view/67211.

Stanislawski, L. V., Kronenfeld, B. J., Buttenfield, B. P., Shavers, E. J., 2023. At what scales does a river meander? Scale-specific sinuosity (S3) metric for quantifying stream meander size distribution. *Geomorphology*, 436, 108734. https://www.sciencedirect.com/science/article/pii/S0169555X2300154X.

Wang, D., Qian, H., 2023. Graph neural network method for the intelligent selection of river system. *Geocarto International*, 38(1), 2252762. https://doi.org/10.1080/10106049.2023.2252762.

Xiao, T., Ai, T., Burghardt, D., Liu, P., 2025. Map Generalization Method Supported by Graph Convolutional Networks. *AGILE: GIScience Series*, 6, 1–10. https://agile-giss.copernicus.org/articles/6/13/2025/agile-giss-6-13-2025.html.

Xiao, T., Ai, T., Burghardt, D., Liu, P., Yang, M., Gao, A., Kong, B., Yu, H., 2024. A road generalization method using graph convolutional network based on mesh-line structure unit. *Geocarto International*. https://www.tandfonline.com/doi/abs/10.1080/10106049.2024.2413549.

Yan, X., Min, Y., Ai, T., 2023. Deep learning in automatic map generalization: achievements and challenges. *Geo-spatial Information Science*, 0(0), 1–22. https://doi.org/10.1080/10095020.2025.2480815.