National College of Ireland

Enhancing Dublin's Bike-Sharing Network: Leveraging Deep Learning for Growth and Efficiency

MSc Research Project
Data Analytics

Valentina Bernal Gomez
Student ID: 23395745

School of Computing
National College of Ireland

Supervisor: Furqan Rustam

National College of Ireland
Project Submission Sheet
School of Computing

Student Name:	Valentina Bernal Gomez
Student ID:	23395745
Programme:	Data Analytics
Year:	2025
Module:	MSc Research Project
Supervisor:	Furqan Rustam
Submission Due Date:	15 September 2025
Project Title:	Enhancing Dublin's Bike-Sharing Network: Leveraging Deep Learning for Growth and Efficiency
Word Count:	7918
Page Count:	25

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature:	Valentina Bernal Gomez
Date:	15th September 2025

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST:

Attach a completed copy of this sheet to each project (including multiple copies).	☐
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	☐
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	☐

Assignments that are submitted to the Programme Coordinator office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Enhancing Dublin's Bike-Sharing Network: Leveraging Deep Learning for Growth and Efficiency

Valentina Bernal Gomez
23395745

Abstract

The population around the world is growing every year , which makes more people move into main cities, for different reasons such as work, studies or tourism, and traffic is getting worse, as most people use cars or public transport. Bicycles are becoming a healthy and sustainable option to reduce congestion and move faster inside the cities. For this reason, bike-sharing services have been introduce in large cities to encourage people to use bikes and reduce traffic conditions. However, when focusing in planning and traditional approaches there are a lack of use of spatial and temporal analysis and fail to address class imbalance in the data use. This research aims to improve and help with the expansion of Dublin's bike-sharing network by developing a deep learning based methodology that combines clustering techniques, spatio temporal analysis, and predictive modeling using Graph Neural Networks, furthermore, the study includes an analysis of the network infrastructure and provides strategic suggestions for system improvement. Multiple variations were tested and the best performance was achieved by a GCN model with focal loss and no leakage information, reaching an accuracy of 0.986 and a F1-score of 0.971. The findings demonstrate the potential of integrating deep learning models into urban planning to improve the network and the efficiency of the system.

Keywords: Clustering Techniques, Predictive Modeling, Spatio-temporal Analysis, Bike-Sharing Systems.

1 Introduction

Since COVID-19, people are more conscious about their mental and physical health, this has encouraged them to exercise more often and use healthier ways of transport, like the bike, which is also a sustainable transport. Cycling release endorphins and reduce stress. As mentioned in the study by Ma et al. (2021), "regular cycling was negatively associated with psychological distress and positively associated with life satisfaction." Cycling also allows people to move around the city and save time on their journeys.

Dublin, is a big city and traffic is one of the main challenges and problems; people are spending long hours on the road, leading to lose time and productivity. McCárthaigh (2023) highlights that congestion in Dublin has big economic costs and it is expected to rise above €1.5bn by 2040. This problem makes people prefer to cycle rather than take a bus, especially because in some areas bus services are not available during late hours, even having this possibility of cycling bike users sometimes struggle to find free docking spaces, for example, if stations A is full, they have to leave the bike in a farther station

B. This usually happens because station A is more convenient for local residents and it depends on demand and usage patterns of each station.
In addition, bike-sharing services align with Sustainable Development Goals (SDGs) such as goal 13 Climate action and 11 Sustainable development. Unlike non-electric cars that contribute to global warming, bicycles are a clean and sustainable option, the construction of more lanes exclusive for the use of bikes will help with goal 11, since cycle is affordable, healthy and good for a sustainable economy. It also reduces noise pollution because fewer cars are on the roads.
Artificial intelligence is being applied more often in mobility planning. Graph Neural Networks (GNNs) have shown effectiveness in modeling transportation systems, due to it learn spatio-temporal features and recent studies, such as Liang et al. (2023) demonstrate that those models outperform traditional prediction methods.

Research Question What impact can the application of clustering and graph neural network algorithms have on optimizing the expansion strategy of Dublin's bike-sharing system and how the integration of these approaches can improve efficiency and customer satisfaction?

Main contributions of this research

- Proposes the combination of clustering techniques with GNNs to predict and suggest an optimal expansion of the system. This combination helps to identify high-demand areas with a needed of an expansion of the network.
- Include spatiotemporal, socio-demographic data to enrich the model and have a better prediction. The model captures complex urban features when implementing variables such as commuting modes, age and employment status, reflecting the real world demand conditions.
- Compare different variations of GNNs to help with the efficiency of the network. Different architectures of GNN were implemented including GCN, GraphSAGE and GAT, those models offer valuable insights for expansion planning.
- Supports a global goal for sustainable development, encouraging bike use. This research contributes to reduce carbon emissions, traffic congestion and improve urban mobility as this mode of transport is more accessible and sustainable.

Structure of the thesis

The thesis begin with section 2 which presents the related work, illustrating key studies in bike-sharing systems, spatio-temporal analysis, and deep learning applications. Section 3 demonstrates the methodology, including dataset preparation, clustering, and GNN model implementation. Section 4 shows the design specification and the system architecture with de variations of the model implemented in this research. Section 5 presents the system configuration and implementation pipeline. Section 6 illustrates and discusses the experimental results. Finally, section 7 discusses the conclusions and future work.

2 Related Work

The use of those systems (bike-sharing services) has grown a lot in the last years and has become an important part of their daily transport. This system reduce traffic congestion and mitigates the impact of climate change. Expanding the bike network brings different benefits, especially in cities like Dublin with high population, where more people move to the country. As said before, traffic in Dublin is a serious problem and during rush hours the travel time can increase a lot. Bicycle-sharing programs also reduce congestion and encourage the use of public transport, working as a complementary option for mobility Campbell (2023). This section illustrates the most relevant studies about bike-sharing services where traditional methods, machine learning based approaches, deep learning based methods and hybrid or other relevant approaches are presented.

2.1 Traditional Approaches

Traditional methods have been used for a long time to analyze spatial and temporal patterns in bike-sharing networks, including clustering and regression models. For instance, Lee and Leung (2023) analyze the bike-sharing services and their relationship with neighborhood characteristics to improve the integration of the system with the infrastructure. They include clustering based on time series ridership to group bike stations according to demand patterns. Weighted dynamic time warping was used to assign a larger weight to larger time differences. Eight clusters of bike stations were identified, and the authors observed that stations close to important points of interest such as public transport, restaurants, and shopping centers have high demand.

Whereas the study by Lee and Leung (2023) includes different points of interest, the study by Xin et al. (2023) analyzes bike mobility by reconstructing bike mobility chains (BMCs), grouping stations with similar patterns using word embedding and clustering. Moreover, Xin et al. (2023) found that the stations close to train stations have a net flow regardless of weekdays or holidays, and that stations outside Manhattan require incentives, as people in these areas do not use bikes often. The first study provides valuable results that will support the present research in planning new stations by considering different points of interest. Similarly, the second study highlights the spatiotemporal analysis to understand how bike mobility varies depending on the day of the week and studies the connectivity between stations and the distribution of bikes in different urban areas. Both studies will contribute to optimizing the station distribution and improving bike infrastructure by leveraging spatiotemporal analysis from different perspectives.

Furthermore, Zhan et al. (2023) include in their analysis important variables that influence the usage of dockless bike-sharing and metro systems in Beijing. They highlight the relevance of bike density and metro ridership around stations. Their study shows how, after certain thresholds of bike density and metro ridership are surpassed, important effects emerge in access and egress trips. Although they studied the dockless bike system, their findings can be applied to the development of the expansion of Dublin's BSS, as Zhan et al. (2023) imply that stations located in areas with a high concentration of metro and bike users are more efficient. These findings support the idea that the relocation or creation of new stations should consider not only user density but also proximity to other modes of public transport, such as metro stations or, in the case of Dublin, DART or Luas stations.

On the other hand, spatiotemporal analysis can be combined with clustering techniques

to improve model performance. These techniques are important for identifying recurrent patterns in clusters and supporting decision-making. For instance, Sun et al. (2023) propose a framework that addresses the rebalancing problem in free-floating bike-sharing services by combining clustering techniques and spatiotemporal demand analysis. The authors introduced 'self-balanced' clusters to rebalance nodes in spatially grouped areas with similar redistribution patterns. In addition, they considered temporal dynamics using a multi-period synchronous rebalancing method. The results demonstrated the relevance of combining these two methodologies in reducing costs and improving system efficiency.

2.2 Machine Learning-Based Approaches

Different studies have applied machine learning methods such as Support Vector Machine (SVM), K-Nearest Neighbors (KNN) and Random Forest to analyze bike-sharing system patterns. For example, Lee and Leung (2023) implemented machine learning models such as SVM and KNN were applied to predict which group each station belongs to based on its urban characteristics. In the same way, Mohseni et al. (2025) demonstrated in their study that the GNN model outperformed traditional methods such as XGBoost and Random Forest. Additionally, Lei et al. (2024) found that this model outperformed Random Forest in predicting the number of floors in the building.
An study made by Zhou et al. (2025) used methodologies like decision tree and boosting models to classify bike-sharing trips as commuter or non-commuter, the research included trip duration, stations-level data and temporal patterns and demonstrated that the prediction accuracy improved when combining the two approaches.

2.3 Deep Learning-Based Approaches

Deep learning models, specially Graph Neural Networks (GNNs), have demonstrated optimal results in modelling spatial and temporal studies in bike-sharing services. Liang et al. (2023) used a Graph Neural Network (GNN) to predict station-level demand based on urban environmental data. Through the construction of multiple graphs centered around each station, the model utilizes attention mechanisms to learn spatial correlations between stations and build environmental factors, improving the accuracy of demand predictions. The study shows that this approach outperforms traditional spatial regression models and other machine learning models. This study is relevant for the expansion of the bike-sharing system in Dublin, as it provides an idea to identify optimal locations for new stations, not just by considering the demand for existing stations but also environmental features and correlations between stations.
Tang et al. (2022) presents a hybrid deep learning model with irregular convolution and LSTM to improve short-term bike-sharing demand forecast in different cities, the study include irregular time intervals and it significantly shows a better performance over traditional LSTM and CNN models. Combining a deep learning model and a spatiotemporal analysis might improved the model performance. For instance, Zhang et al. (2023), predict bike flows by using a deep learning model based on structural causal mechanisms and demonstrated that the model reduces spurious correlations and outperforms traditional deep learning models when capturing causal relationships in spatiotemporal data.
Lei et al. (2024) included GraphSAGE in their study aiming to predict building characteristics and urban scale, including street-level context across three cities. The authors

Their results confirm that GraphSAGE is a generalizable model useful for urban prediction tasks. The benefits that this architecture brought in the study of Lei et al. are similar to the prediction of new bike-sharing stations, as it enables making predictions at unseen candidate locations enriched with sociodemographic data.

In their study, the authors focused on predicting station-to-station ridership for network expansion using the Graph Sample and Aggregate framework, which includes GNNs to generate node embeddings, capturing spatial relationships and interactions between stations. This model incorporates sociodemographic and station data to improve accuracy. The study was tested on the Bikeshare Toronto dataset, and the results concluded that GNN outperformed traditional methods like XGBoost, linear regression, spatial regression, and artificial neural networks. Moreover, Mohseni et al. (2025) consider urban core connectivity, suburban access, transit integration, equitable accessibility, and tourist hubs, which help in the strategic location of new stations. This model supports the idea that GNNs provide essential insights to identify new station locations, considering both the existing network and various urban factors.

Bike-sharing services require a thorough approach to determine new locations for bike stations. Some recent studies have used optimization algorithms to identify the best locations for additional bike stations, considering features such as proximity to residential or commercial areas and bus or train points. Bike-sharing systems (BSS) are a key element in sustainable cities. Several studies focus on system optimization through the analysis of spatiotemporal patterns. In this context, Roantree et al. (2024) built a geospatial graph using data from Moby Bikes (a dockless bike system used in Dublin), identifying strategic locations for future stations. Furthermore, the Louvain algorithm detects usage communities with similar spatiotemporal dynamics, which helps improve redistribution efficiency. Graph networks have been used to understand and model the structure of the bike network. Similarly, Cuong et al. (2024) propose a graph network approach to model activity in transport networks such as bike-sharing services. The use of graph databases allows for the interpretation of mobility dynamics and the discovery of essential information to locate new stations. The results obtained improve the possibility of using graph networks in the expansion of Dublin's BSS. In addition, Xiao et al. (2023) demonstrated that GNNs performed 8% better than Artificial Neural Networks (ANN) when predicting travel demand between stations. The results show that the information of the local network is essential in the planning of BSS, for example, whether a station has few or many points of interest, such as schools or shopping centers. Consequently, this model supports design decision-making in the system, such as where new stations can be located using prior information.

Likewise, Liu, Jiang, Zhou, Kwan et al. (2023) proposed DST-GraphSAGE, a dynamic spatiotemporal model for traffic flow prediction. The model captures dynamic spatial dependencies and long-term temporal trends using a spatial-temporal GraphSAGE module, and incorporates attention mechanisms and dilated convolutions. Their results demonstrated improved performance over baseline methods. This reinforces the importance of GraphSAGE in modeling complex urban systems, as it generalizes to unseen nodes enriched with sociodemographic and geographic features. In addition, Zhao et al. (2024) proposed the GDLBE model (Graph Deep Learning with Built Environment), which also integrates GraphSAGE to predict short-term bus travel demand. GraphSAGE was used to capture spatial-temporal dependencies, allowing the model to embed nodes (e.g., bus stops) that were not seen during training. Moreover, features were aggregated from neighboring nodes to learn spatial context and temporal dynamics. The results showed that

this model outperformed baseline methods in an experiment conducted in Shenzhen.

2.4 Hybrid or Other Relevant Approaches

Hybrid approaches combine different techniques and use new model methods such as custom loss function. Focal Loss is often used in classification tasks to address class imbalance. When implemented, it allows the model to focus more on misclassified samples from the minority class. For instance, Sun et al. (2023) used spatiotemporal analysis and clustering to help the model with the imbalance problem. Moreover, Liu, Lu, Chen, Fong, Ma and Zhang (2023) evaluated crash risk in urban road networks and included Focal Loss to address the problem of class imbalance in the distribution of crash occurrences. The results showed an improvement in model performance compared to traditional methods, as the model focused on high-risk crash events that represent the minority class.
Some studies also optimize the system of bike-sharing services, Caggiani et al. (2024) in their study combine a reactive and real-time task for repositioning bikes in the network. The model use a time-sensitive demand and matches bikes with redistribution tasks to enhance the availability of bikes and minimize operational costs. Similarly, Focal Loss was implemented in a financial application where class imbalance was also present. Humranan and Supratid (2023) applied a GCN model with Focal Loss to detect illicit Bitcoin transactions. In this case, the illicit nodes were the minority class, which caused a prediction challenge. Their results proved that Focal Loss improved the model performance by reducing the rate of misclassifications in an imbalanced class.
These findings support the implementation of Focal Loss in this research, as it also involves a class imbalance problem. The use of Focal Loss helps the model focus on the minority positive class, improving both precision and recall. Other studies like the one conducted by Chen et al. (2024), where they implemented a distribution and similarity analysis to study the differences of usage patterns between weekdays and weekends in six different cities using Jensen-Shannon divergence. The results highlight the importance of spatiotemporal analysis in demand patterns.

Table 1: Summary of related studies in bike-sharing, mobility and class imbalance prediction.

Author/Year	**Method**	**Dataset**	**Results**	**Key Limitation**
Lee and Leung (2023)	Clustering, SVM, KNN	Bike stations + Points of interest	Demand was influenced by points of interest	No predictive modeling
Xin et al. (2023)	Word embedding, clustering	Manhattan bike-sharing services	Chain patterns change every day	No spatial analysis included
Zhan et al. (2023)	Threshold regression	Beijing metro and bikes	Illustrated metro-bike usage	No comparison made

Author/Year	Method	Dataset	Results	Key Limitation
Sun et al. (2023)	Clustering, spatiotemporal analysis	Urban bike trips	Supports rebalancing	No predictive modeling
Liang et al. (2023)	GNN	Urban environmental data	Outperforms regression	No expansion planning
Mohseni et al. (2025)	GraphSAGE, sociodemographics data	Toronto bike-sharing services	High accuracy	Findings not generalized
Roantree et al. (2024)	Spatial graph, Louvain	Dublin Moby data	Cluster detection in Dublin	No predictions with GNN
Cuong et al. (2024)	Spatio-temporal graph	Multi-mode transport	Mobility improved	No predictive modeling
Xiao et al. (2023)	GNN and ANN	Transport network data	GNN has 8% higher accuracy	Not applied to bike-sharing
Lei et al. (2024)	GraphSAGE, street view data	Urban mapping data	Generalized to more cities	Not focused on bike-sharing
Liu, Jiang, Zhou, Kwan et al. (2023)	DST-GraphSAGE and attention	Traffic datasets	Shows dynamic flow prediction	Not focused on bike-sharing
Zhao et al. (2024)	GraphSAGE and built environment	Bus travel demand	Predicts short-term demand	Not focused on bike-sharing
Liu, Lu, Chen, Fong, Ma and Zhang (2023)	ASTGCN and Focal Loss	Urban crash risks	Improved rare event detection	Not focused on bike-sharing
Humranan and Supratid (2023)	GCN and Focal Loss	Blockchain fraud data	Reduces errors for minority class	Not focused on bike-sharing
Zhou et al. (2025)	ML classification	Commute trip data	Detects commuter use	No deep learning applied
Chen et al. (2024)	Spatio-temporal stats	Bike-sharing systems	Identifies usage variation	No predictive modeling
Zhang et al. (2023)	Neural causal model	Urban bike flow	Reveals causal structure	Complex input required
Caggiani et al. (2024)	Real-time assignment	Bike-sharing rebalancing	Faster response	No deep learning used

Author/Year	Method	Dataset	Results	Key Limitation
Tang et al. (2022)	IrConv and LSTM	Short-term forecasts	Boosts forecast accuracy	Needs long sequences
Ma et al. (2021)	Causal inference	Transport surveys	Bike use linked to wellbeing	No predictive modeling

2.5 Limitations and Research Gaps

The lack of integration of sociodemographic and spatiotemporal data is a common limitation, as many studies rely on historical usage or weather data without integrating these important features to enrich the predictive modeling, this can reduce the model performance by failing to capture complex urban information. Furthermore, other research that includes Graph Neural Networks focuses on demand prediction without including expansion planning, which is a contribution of this research.

For instance, Liang et al. (2023) used a GNN to predict demand with environmental data but did not consider socio-demographic information. Moreover, Lee and Leung (2023) studied the correlation between station demand and the points of interest such as public transport, restaurants or shopping centers, but did not address expansion planning for the network. Xin et al. (2023) analyzed bike mobility chains and used clustering to study flows in Manhattan, however, the study was limited to spatio-temporal connections and they did not include expansion planning. Roantree et al. (2024) built a geospatial graph for another bike system in Dublin (Moby Bikes) identifying strategic locations, but this work remained exploratory and they did not include socio-demographic enrichment.

Another gap is the that recent studies lack of comparative analysis of GNN variations, those studies rely on a single model without evaluation alternative approaches. This research addresses these limitations by combining spatiotemporal and sociodemographic features into the model, applying multiple GNN variations and focus on expansion planning.

In addition, Zhan et al. (2023) studied dockless bike sharing and metro systems in Beijing, where they highlight the importance of ridership and density thresholds, but this study did not include socio-demographic variables. Furthermore, Sun et al. (2023) combined clustering techniques and spatio-temporal analysis to address the rebalancing problem, but they used it to help with the efficiency in the redistribution rather than an expansion planning. These examples demonstrate that existing studies focuses on demand forecasting or rebalancing but did not combined socio-demographic enrichment, comparative of GNN variations or expansion strategies, which are the research gap for this study.

3 Methodology

This study involved exploratory spatial analysis, clustering, and deep learning models based on Graph Neural Networks to predict new locations for a bike-sharing system in Dublin. Clustering techniques were used in different studies to group urban areas with similar demand patterns, for example, the study made by Sun et al. (2023) combined clustering techniques and spatiotemporal demand analysis to rebalance nodes in spatial

grouped areas with similar redistribution characteristics. Moreover, Lee and Leung (2023) include clustering techniques to group bike stations according to similar demand patterns. In addition, deep learning models and particularly GNNs have been efficient in studies, showing that they outperform traditional approaches, for instance, Mohseni et al. (2025) demonstrated that XGBoost and Random Forest fall behind a GNNs methodology. Another study that supports the used of GNNs is the one made by Xiao et al. (2023) were they demonstrated that this methodology surpassed in 8% an Artificial Neural Network when predicting demand between stations.
Based on this enriched information, different GNN architectures are trained to identify suitable new potential bike stations. To address class imbalance and improve predictive performance, different training strategies are implemented, including batch and focal loss. The trained models include variations of GCN (Graph Convolutional Network), GraphSAGE, and GAT (Graph Attention Network). Finally, model performance is assessed using a range of evaluation metrics such as accuracy, precision, recall, F1-score, and ROC-AUC. Figure 1 outlines the full methodology pipeline employed in this study.

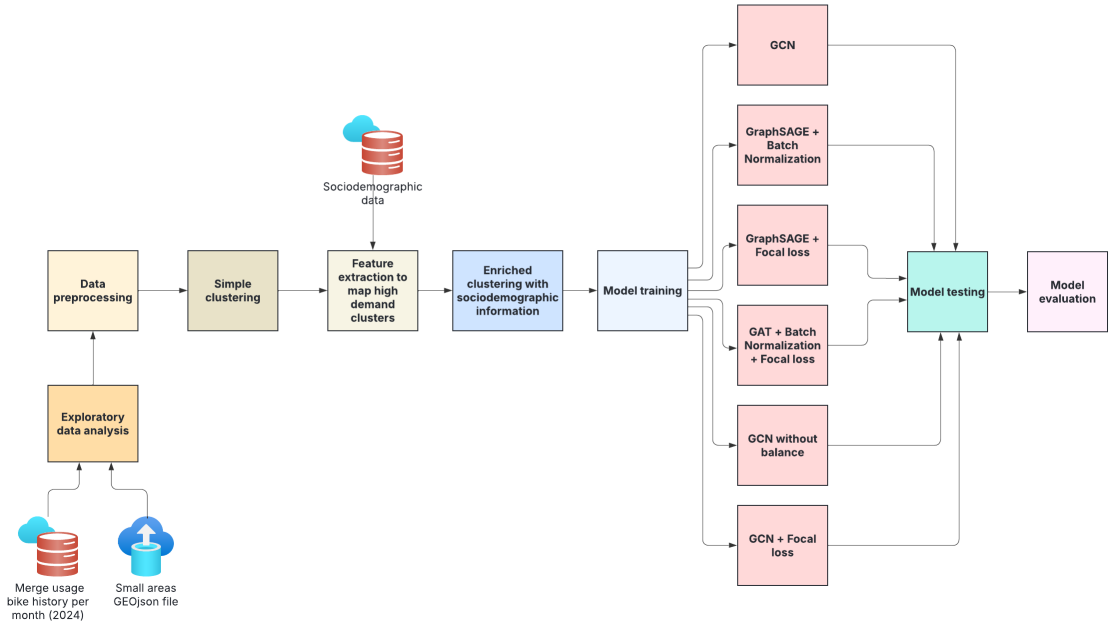

Figure 1: Research pipeline used to identify new bike-sharing station locations in Dublin.

3.1 Dataset Description

To develop the predictive model, multiple datasets were integrated, including historical data on bike station locations from the year 2024, trip records, and usage patterns from Dublin's public bike system, similar to the study made by Xin et al. (2023), where they include an analysis of spatiotemporal characteristics to understand bike mobility outside Manhattan, in addition, Liang et al. (2023) include environmental data to predict demand from learning spatial correlations between stations. Geospatial layers containing the geometries of Dublin, this file enables the mapping of stations and census and demographic data which was extracted from the Central Statistics Office (CSO), including population density, age distribution, commuting behavior, employment, and income levels by Small Area (SA), following approaches like Lee and Leung (2023) where they

analyze the bike-sharing service and their relationships with socio-demographic features or the research made by Zhan et al. (2023) where they include information of bike density and metro ridership around stations in Beijing for the study of how this variables affected the bike-sharing system.
These datasets were merged by spatially joining each bike station with its corresponding small area, the final dataset contains 115 real stations with valuable information such as demographic and infrastructural characteristics. In addition, a set of synthetic candidate stations was generated to simulate potential locations for new stations. These negative samples (label = 0) were created by jittering and combined with the characteristics of nearby real stations to preserve contextual consistency. The final dataset used for model training, testing and evaluation consists of a balanced and enriched data of real and candidate stations.

3.2 Data Preprocessing

Data preprocessing was essential to ensure consistency and quality between sources. This process includes collection, cleaning and transformation of data to prepare it for clustering and model training. First, all datasets, bike-sharing data geoJSON file and socio-demographic data were merged using spatial operations to assign each bike station to its small area. To validate the number of clusters, Elbow Method and Silhouette Score were implemented, the former suggests the best number of clusters, on the other hand, the Silhouette Score measures the cohesion and separation of clusters, based on these metrics, K=3 was selected as the most optimal number of clusters.
The final dataset consisted of enriched nodes representing both existing stations and potential new locations, each described by a consistent set of spatial and sociodemographic features. For GNN training and testing, synthetic candidates stations were created using jittering and enriched with the same features as the real stations, and a binary label was assigned to each location (real = 1, 0 = synthetic) to support supervised learning.

3.3 Clustering

Clustering was applied after evaluate with the Elbow Method and Silhouette Score the best number for clusters, these clusters help us identify groups of bike stations with similar characteristics and similar patterns, moreover, to point out zones with high and low demand. K-means was chosen because it is computationally efficient for large datasets same as the one used in this study after including enriched information. One advantage of this clustering algorithm is that runs efficiently on high dimensional datasets, which was the case for this research given the integration of sociodemographic data GeeksforGeeks (2024). Moreover, this technique can be validated through the Elbow Method and the silhouette score to check the number of clusters.
K-Means was applied first to the enriched dataset without any evaluation, after this Elbow method and silhouette score were applied to validate the number of clusters, and both indicate that k=3 was the most suitable to use. K-Means was applied to the full dataset of 115 real stations assigning each station to one cluster. These clusters give important information to interpret the demand patterns across Dublin.
The K-Means algorithm suggests 3 clusters, each station was assigned to a cluster. The results indicated that:

Cluster 0 represents high-demand stations, it has the highest occupancy rate (37.5%),

meaning that the stations are used frequently, and it has the highest availability (12.28 bikes).

Cluster 1 are underutilized stations, it has the lower capacity (10.26 spots) and low occupancy rate (31.7%), likely stations that are not used much, could be a potentially oversupply.

Cluster 2 are medium-small moderate usage stations, (10.64%) available bikes and moderate occupancy (36.5%), these may be medium-small demand stations in smaller neighborhoods or less central areas.

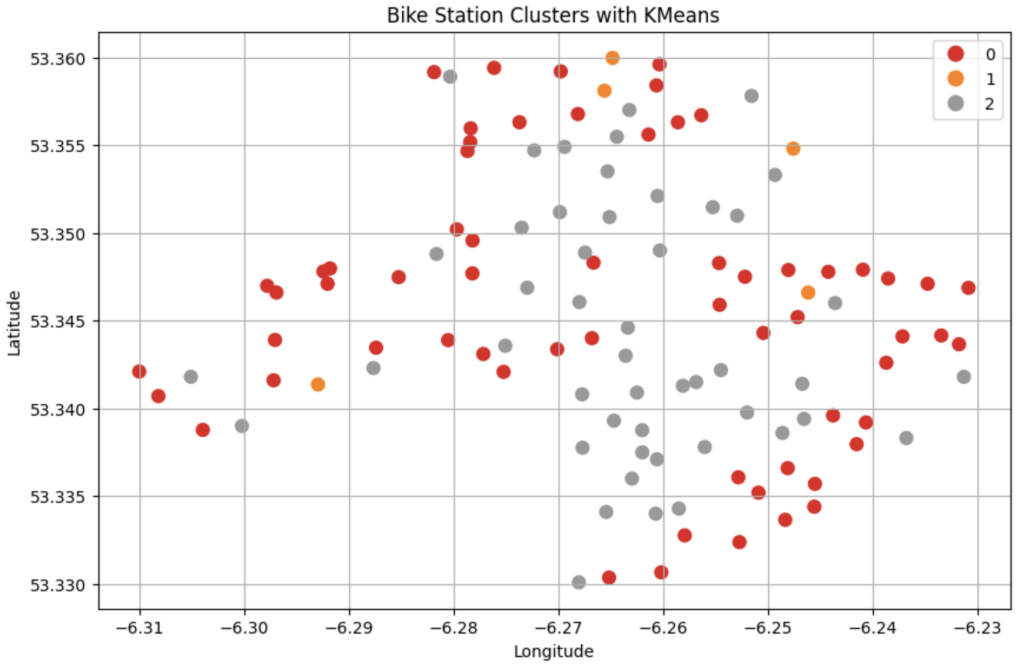

Figure 2: Spatial distribution of bike stations by cluster (K = 3). Cluster 0: city center, Cluster 1: residential, Cluster 2: peripheral areas.

3.4 Graph Construction

To prepare the data for the graph construction that is critical for the Graph Neural Networks (GNNs), each node represents a real bike station or a synthetic candidate location that was created by jittering. Each node includes sociodemographic information, usage data and spatial context.
Edges were constructed using the K-NN algorithm, for each node, we used the optimal K-NN to detect spatial proximity and influence between locations. After jittering, the labels were classified as 1 = real bike stations/positive class, 0 = synthetic candidate locations/negative class. The final graph contains 230 total nodes and 115 real stations.

3.5 Modeling

This study included different Graph Neural Network (GNN) architectures to predict the most suitable candidate locations for new bike-sharing stations. To train the models, a node-level classification was used, where the label indicates whether a node represents a real station (label = 1) or a synthetic candidate location (label = 0). The models differ

in architecture, regularization techniques and loss functions. All of them were trained using Adam optimizer with a learning rate of 0.01. The following architectures were implemented:

Original GCN (Baseline) a baseline Graph Convolutional Network using ReLU activation and dropout of 0.3 to reduce overfitting. The class imbalance was handled using class weights of 0.4 and 0.6 to slightly favor the minority class (real stations).

GCN with Focal Loss (No information Leakage), this model used focal loss ($\alpha = 0.25$ and $\gamma = 2.0$) which is useful for imbalance datasets and the test nodes were disconnected from the graph to avoid information leakage.

GCN without Balance, same GCN architecture as the baseline, but trained on the imbalanced dataset without applying class weights or focal loss. A dropout of 0.6 was applied to avoid overfitting, this model was used to compare the effect of imbalance handling.

GraphSAGE with Focal Loss A 3-layer GraphSAGE model with focal loss. It performed better than its batch-normalized variant.

GraphSAGE with Batch Normalization a 3-layer GraphSAGE model with batch-normalization.

GAT with BatchNorm and Focal Loss a 3-layer Graph Attention Network model with both batch normalization and focal loss($\alpha = 2.0$ and $\gamma = 2.0$), moreover, includes multi-head attention.

3.6 Evaluation Methodology

Model evaluation was performed using a combination of metrics to ensure robustness:

Accuracy: Measures the degree to which data show correctness.

Precision: Measures how many of the true positive predictions were actually real stations.

Recall: Measures the ability of the model to detect all positive bike stations.

F1-score: Provide a balanced view of performance between precision and recall.

This dataset was split into a 80-20 train/test, ensuring that test node were separated from the graph during training. Focal loss was useful and applied to handle imbalance classes, this loss function helps to improve precision and recall.

4 Design Specification

The overall design of this research includes spatial clustering, graph construction and the integration of different Graph Neural Network algorithms to predict new bike-sharing stations in Dublin. The entire pipeline was designed to improve spatial relationships and enriched features to increase the performance for prediction accuracy and model generalization.

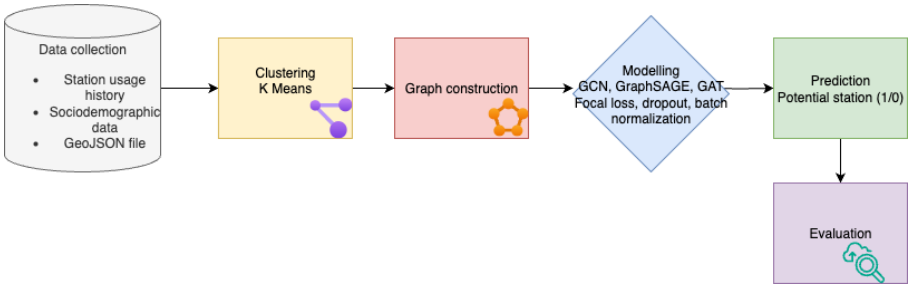

Figure 3: System architecture for predicting bike sharing station candidates

4.1 System Architecture

The system architecture consists of the following items:

Data Collection: The enriched dataset combines data on bike station usage with demographic and geospatial characteristics.

Clustering: KMeans clustering was applied to the real stations to identify demand patterns.

Graph Construction: A spatial graph structure was built, and using the K-Nearest Neighbors, a matrix was created where each node is connected to its closest K nearest neighbors.

GNN Modeling: Three architectures (GCN, GraphSAGE and GAT) were implemented and some variations were included, for instance, focal loss was used in order to address class imbalance, dropout and batch normalization were tested to improve generalization.

Prediction: The results of each model were used to predict and classify whether a node was a potential station or not.

4.2 GNN Model Variants

Six variants of GNN models were trained and tested:

Original GCN (baseline): a two-layer Graph Convolutional Network with 16 hidden units, ReLU activation was selected because of its efficiency in deep learning methodologies and preventing vanishing gradients, dropout of 0.3 to reduce overfitting and the final output layer use for classification was log-softmax. The class imbalance was handled using class weights of 0.4 and 0.6 to slightly favor the minority class (real stations).

GCN + Focal Loss (No Leakage): a two-layer GCN architecture, a dropout of 0.6 to avoid overfitting. This model used focal loss (alpha=0.25 and gamma=2.0) which is useful for imbalance datasets, so the model focuses on misclassified classes like this research, where we have real and synthetic stations. In addition, test nodes were disconnected from the graph to avoid information leakage which is a common problem in GNNs due to embedding propagation can leak test information into training.

GCN (No Balance): same GCN architecture as the baseline, but trained on the imbalanced dataset without applying class weights or focal loss. A higher dropout of 0.6

was applied to avoid overfitting due to the model tend to overfit when the imbalance is not addressed, this model was used to compare the effect of imbalance handling.

GraphSAGE + BatchNorm: a 3-layer GraphSAGE model with 64 hidden layers this number was changed over the process to see how the model performed, batch-normalization was included because stabilize training. Leaky ReLU was used instead of normal ReLU because it can help to prevent the "Dying ReLU" problem Rallabandi (2021), and these techniques where combined to compare which one performs better.

GraphSAGE + Focal Loss: a 3-layer GraphSAGE model with focal loss, 32 hidden units per layer and LeakyReLU activation were used to prevent the "Dying ReLU" problem Rallabandi (2021), this GNN architecture aggregates better neighborhood information and help to unseen nodes. Focal loss (alpha=1, gamma=2) was included to give more relevance to misclassified stations, this helps the model with the imbalance issue between real and synthetic stations. Dropout was applied to improve generalization and avoid overfitting. This model performed better than its batch-normalized variant.

GAT + BatchNorm + Focal Loss: a 3-layer Graph Attention Network model, with 32 hidden layers, this number was changed over the process to see how the model performed. This model also included batch normalization after each layer to make training more stable and improve learning, additionally, focal loss (alpha = 2.0 and gamma = 2.0), and 2 attention head per hidden layer. GAT was selected to compare with the other models and checked which one works better in the enriched dates, also GAT introduces an idea where not all the neighborhood nodes contribute equally to a new node, this allows the model focus just on the most important neighbors in the prediction tasks Birkins (2023).

5 Implementation

The implementation phase involved the setup of the development system, merging different databases, model training and testing and the prediction of new locations for the bike-sharing system variations of Graph Neural Networks (GNNs).

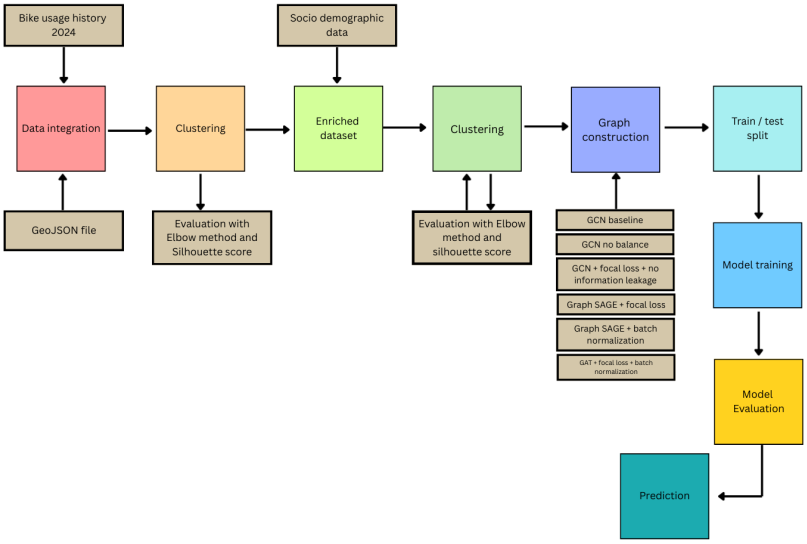

Figure 4: Implementation diagram

5.1 System Configuration

The implementation was develop using a powerful hardware and software setup to ensure the best performance. The research was executed on a system with an Apple M3 processor and 24 GB of RAM, running in a macOS Sequoia (version 15.5). The development environment included Jupyter Notebook for exploratory analysis and Google Collab for training and testing the models, the programming language was Python 3.12.7.

5.2 Libraries and Frameworks

To ensure efficient implementation of the different models, the system included multiple libraries as follows:

Data processing and manipulation: Pandas, NumPy, GeoPandas

Machine learning and evaluation: Scikit-learn

Deep learning and GNNs: PyTorch 2.1, PyTorch Geometric 2.3

Spatial visualization: Folium

Visualization: Matplotlib, Seaborn

5.3 Implementation Pipeline

The implementation pipeline was conducted as follows:

Data collection - After merging the 2024 bike usage history dataset with the GeoJSON file and enriched the data, the final dataset contained 115 rows that corresponded to real stations and included 34 features.

Candidate generation and labeling: Synthetic candidate stations were created after applying jittering to the real stations to simulate locations close to real stations helping the model learn what features distinguish as a good or a bad location. This doubled the dataset to 230 rows where 115 stations correspond to label. And the other 115 synthetic stations correspond to label 0 and maintaining the same feature structure.
Train/Test split: the dataset was splitted into 80% training (184 nodes) and 20% testing (46 nodes) and on the 'no leakage' model test nodes were disconnected from the graph during training to prevent leakage information. The training labels had a shape of 184 x 1 and the testing labels 46 x 1.

Clustering - K Means clustering (K=3) was applied to group groups with similar patterns, this number balances low inertia (compact clusters) and was selected based on the results of the Elbow Method and the Silhouette Score, moreover, was the most justifiable choice to identify different station usage profiles. The clusters were classified in three: high-demand, underutilized and moderate stations.

Graph construction - To create the edge index (graph structure) that connects nodes that are geographically close, each node was defined as a real or synthetic station. The edges were constructed using the K-Nearest Neighbors method,most models used k=5, while GraphSAGE and GAT used k=8. The result was converted into edge index, the format required by PyTorch Geometric to define edges in the graph.

Candidate generation and labeling - It was necessary to generate negative samples (synthetic candidates) by applying a small random jitter to each real bike station location, to create points close to the real stations, this helps train the model to know where not to place a new station. This samples were labeled as 0 and the real stations 1 in order to create a binary classification problem.

Model training - The dataset was divided into 80% training and 20% testing. All models were trained using different loss functions, dropout and hyperparameter. Six variants were implemented independently:

- GCN models: for the three models a 16 hidden layer dimension was fixed, the dropout started with 0.3 for the baseline and increased to 0.6 in the model used with focal loss and no balance to reduce overfitting. The baseline model applied class weights of 0.4 and 0.6 to manage class imbalance, and the version 'no leakage' used focal loss of (α=0.25, γ=2.0) and disconnected test nodes to prevent information leakage. The model GCN without balance was trained without any combined techniques to serve as a comparison.

- GraphSAGE models: the model that combined focal loss was tested with 32 hidden layers, do not included batch normalization and a focal loss of (α=1, γ=2) to improved recall for minority classes, while the other model combined with batch normalization included 64 hidden layers and used LeakyReLU that stabilized training and help with the issue of 'dying.

- GAT model: Included 32 hidden layers and 2 attentions heads per hidden layer. Batch normalization was included to stabilize learning and a focal loss (α=2, γ=2) to help with the imbalance issue.

The hyperparameters used were not chosen arbitrarily, they were tuned through experimentation and combining between them to see which one works better. I adjusted the

number of hidden units, dropouts rates and loss function parameters to observe which settings achieved the best balance in the metrics. Class imbalance was addressed using weighted loss functions or Focal Loss depending on the model.

Hyperparameter configuration - As shown in table 2, different hyperparameters were used in the models:

Model	**Layers & Hidden Units**	**Activation**	**Regularization**	**Loss & Class Balance**	**Graph specifics**	**Additional information**
GCN (Baseline)	2 layers, 16 hidden units	ReLU	Dropout 0.3	Class Weights [0.4, 0.6]	K-NN graph (k=5)	Adam optimizer (lr=0.01), 200 epochs, early stopping.
GCN + Focal Loss (No Leakage)	2 layers, 16 hidden units	ReLU	Dropout 0.6	Focal Loss (α=0.25, γ=2.0); no class weights	K-NN (k=5) + test nodes disconnected	PyTorch Geometric, focal loss custom function. Adam (lr=0.01), 200 epochs. Test nodes disconnected during training.
GCN (No Balance)	2 layers, 16 hidden units	ReLU	Dropout 0.6	Cross-Entropy; no balancing	K-NN (k=5)	Same as baseline but without class weights or focal loss. PyTorch Geometric, 200 epochs, Adam optimizer.
GraphSAGE + Focal Loss	3 layers, 32 hidden units per layer	LeakyReLU	Dropout	Focal Loss (α=1, γ=2); no class weights	Mean aggregation; K-NN (k=8)	PyTorch Geometric GSAGEConv. Adam (lr=0.01), 200 epochs. Focal loss improved misclassified class.
GraphSAGE + BatchNorm	3 layers, 64 hidden units per layer	LeakyReLU	BatchNorm + Dropout	Cross-Entropy; class weights applied	Mean aggregation; K-NN (k=8)	SAGEConv + BatchNorm1d. Adam (lr=0.01), 200 epochs.
GAT + BatchNorm + Focal Loss	3 layers, 32 hidden units; 2 attention heads/layer	LeakyReLU	BatchNorm + Dropout (features + attention)	Focal Loss (α=2, γ=2); no class weights	Multi-head attention; K-NN (k=8)	PyTorch Geometric GATConv. Adam (lr=0.01), 200 epochs. BN after each layer, focal loss during training.

Table 2: Detailed architectures and configurations of implemented GNN models.

Prediction and visualization - After model training, each model was used to predict whether a candidate location was suitable for a new stations, for the output probabilities, different custom thresholds (e.g., 0.3 or 0.45) based on the F1-score were implemented in order to improve classification balance. The predictions were mapped using the library named Folium, where interactive visualization can be observed, real stations and predicted new stations are visualized.

6 Evaluation and Results

This section provides a detailed explanation of the different model performance and results. It highlights the most relevant results and the importance of them in the resolution of the research question.

The final enriched dataset of real stations contained 115 rows and 34 features. Features contained information related to station description and capacity, latitude, longitude and geometry, and usage measures such as average available bikes and occupancy rate per station, and sociodemographic information. The dataset includes 2 integer variables such

as, station id and cluster, 29 floating variables some examples include latitude, longitude, range of ages, commuting modes and employment or education features, moreover, there are 2 categorical variables such as station name and the census small areas and 1 geometry variable that correspond to the spatial location of each station.
There are 115 unique station names that are distributed across 88 small areas around Dublin. After generated candidate stations, the dataset increased the number with 115 corresponding to positive samples or real stations and 115 negative samples or synthetic stations, this created a binary dataset.

6.1 Clustering Results

After choosing the optimal number for K-Means clustering, three clustering were generated based on station features, for example, bikes availability, usage, commuting transport and sociodemographic characteristics. The figures below illustrate the differences between clusters:

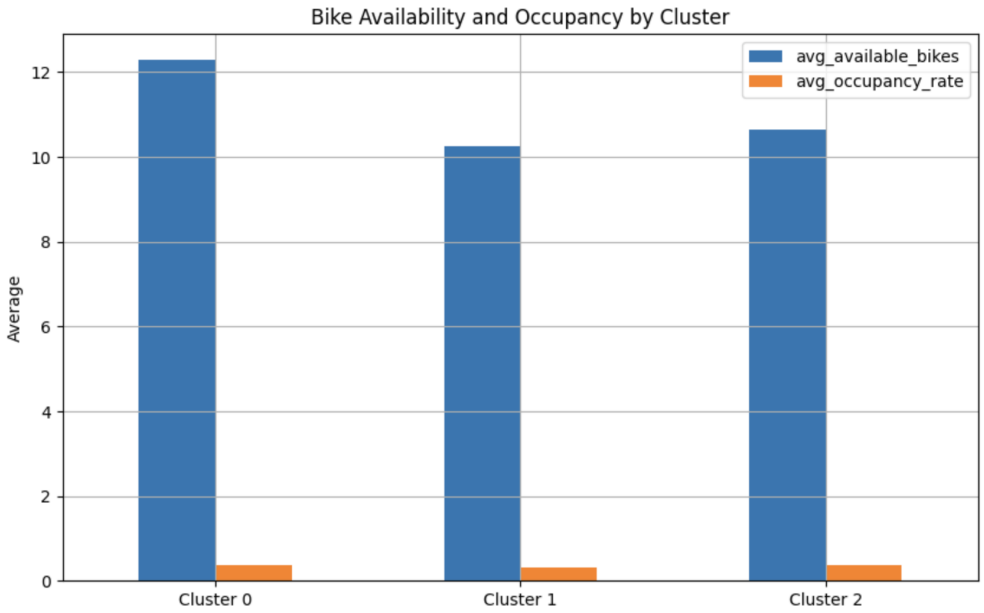

Figure 5: Bike Availability and Occupancy by Cluster

This bar chart illustrates that cluster 0 has the highest occupancy rate (37.5%) and bike availability of 12.28 bikes, indicating stations in high-demand areas with high turnover. Likely to be stations in central or business areas. Cluster 1 has the lowest available bikes (10.26) and lowest occupancy rate (31%) indicating underutilized stations or areas with lower demand. Cluster 2 has moderate bike availability (10.63) bikes and moderate occupancy (36%) indicating regular usage in residential zones.

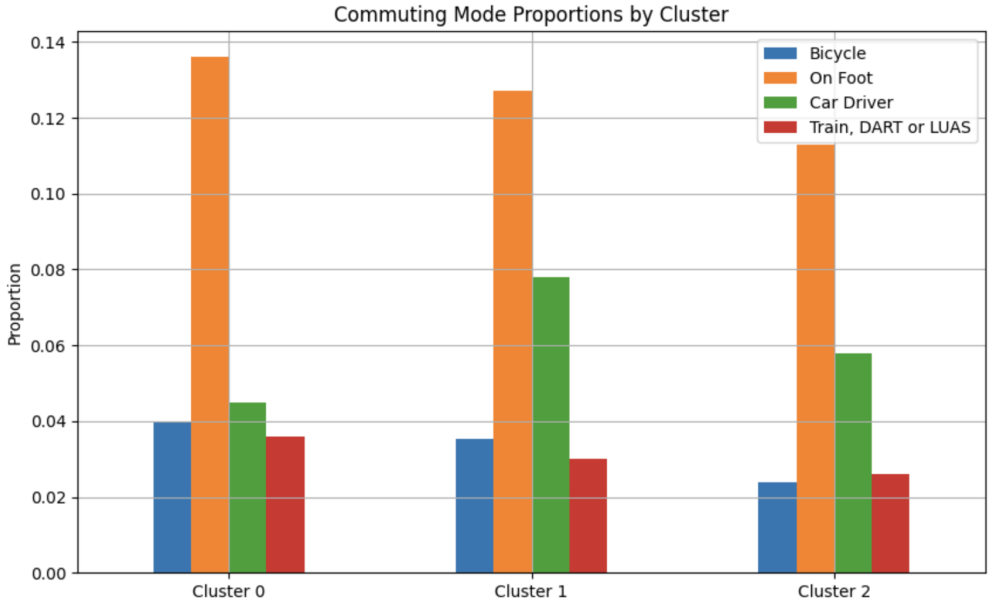

Figure 6: Commuting Mode Proportions by Cluster

This bar chart shows that cluster 0 has a high percentage of use for walking commuters, lower public transport usage and more people commute using bikes, these might indicated a business area with high foot traffic or available bike lanes. Cluster 1 shows the highest car usage percentage, which may indicate suburban areas. Cluster 2 shows a balance between bike usage and public transport access, likely to be a mixed areas with residential and business zones.

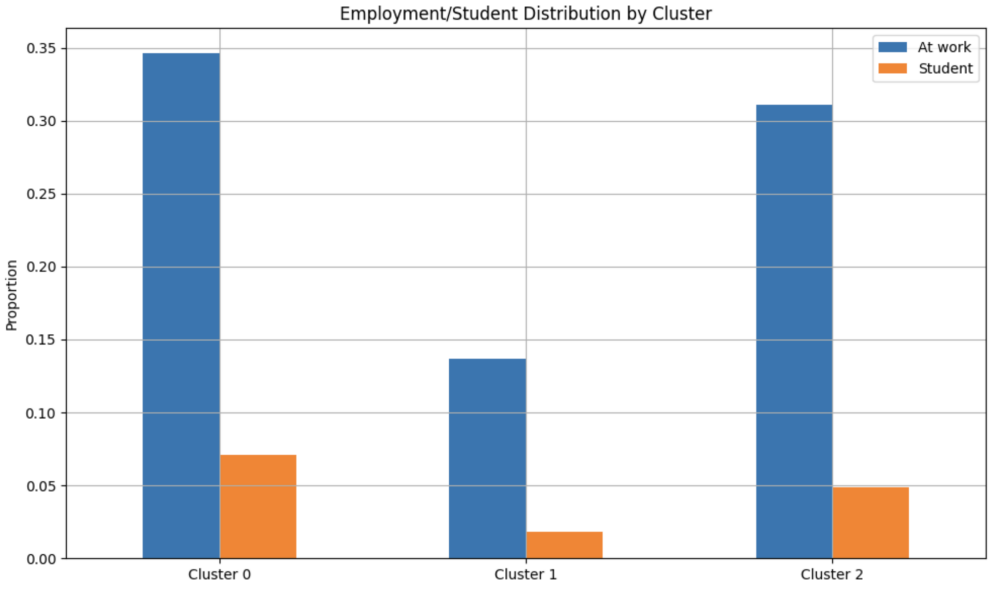

Figure 7: Employment/Student Distribution by Cluster

This bar chart indicates that cluster 0 has the highest proportion of employed people, matching with central zones. Cluster 1 illustrates the lowest employed people and students, matching with underutilized stations. Cluster 2 has a high proportion of working people and moderate student presence.

Overall, these graphics support the relevance of combining the three clusters with model training for stations expansion strategy.

6.2 GNN Model Results

Accuracy, precision, recall and F1-score are widely used to assess classification problems, since this research represents a classification problem, the most appropriate metrics are those ones. Moreover, Sokolova and Lapalme (2009) studied 24 performance metrics and shows that there are complementary, for example, precision indicate the proportion of positive predictions that were correct, recall tell us the proportion of the real positives stations that the model was able to detect, and F1-score gives us a balance metric between recall and precision. In the study of Sokolova and Lapalme (2009) they mentioned that relying just in accuracy might be wrong and it's much better to evaluate models combining several metrics together. This section includes the performance of all models, evaluated with accuracy, precision, recall, and F1-score.

The GCN model with Focal Loss and no information leakage achieved the best performance, with an F1-score of 0.976 and high accuracy 0.986. Confirming that disconnecting test nodes during training can help the model learn information about just test sets, and focal loss assists class imbalance and both together improve generalization. In contrast, GCN without balancing result in a lower performance with a F1-score of 0.471, highlighting the negative impact when class imbalance is not addressed.
GraphSAGE with focal loss showed strong results with a F1-score of 0.622, demonstrating a better performance than its batch-normalized equivalent model with a F1-score of 0.444. The results point out that focal loss had a better effect in this technique than batch normalization. The GAT model was benefit when using focal loss and a multi-head attention, achieving a F1-score 0.647, indicating that the model overpredicts potential stations (precision = 0.489).
Overall, these results indicate that combining different techniques might improve the model performance. In addition, some models were evaluated using a custom threshold to optimize F1-score. The metrics are summarized in Table 3.

Table 3: Performance metrics for each GNN model variant

Model	**Accuracy**	**Precision**	**Recall**	**F1-score**
Original GCN	0.587	0.587	1.000	0.740
GCN + Focal Loss (No Leakage)	0.986	0.944	1.000	0.971
GCN (No Balance)	0.348	0.345	0.741	0.471
GraphSAGE + Focal Loss	0.630	0.636	0.609	0.622
GraphSAGE + BatchNorm	0.348	0.387	0.522	0.444
GAT + BatchNorm + Focal Loss	0.478	0.489	0.957	0.647

6.3 Suggested Stations Visualization

GCN with focal loss and no leakage and GraphSAGE with focal loss were the two best performing models, to visualize the performance, the figures below show the predictions made by them. The green star markers represent the predicted stations suggested by the

GraphSAGE + Focal Loss model, while the green plus markers correspond to the GCN + Focal Loss (No Leakage) model.

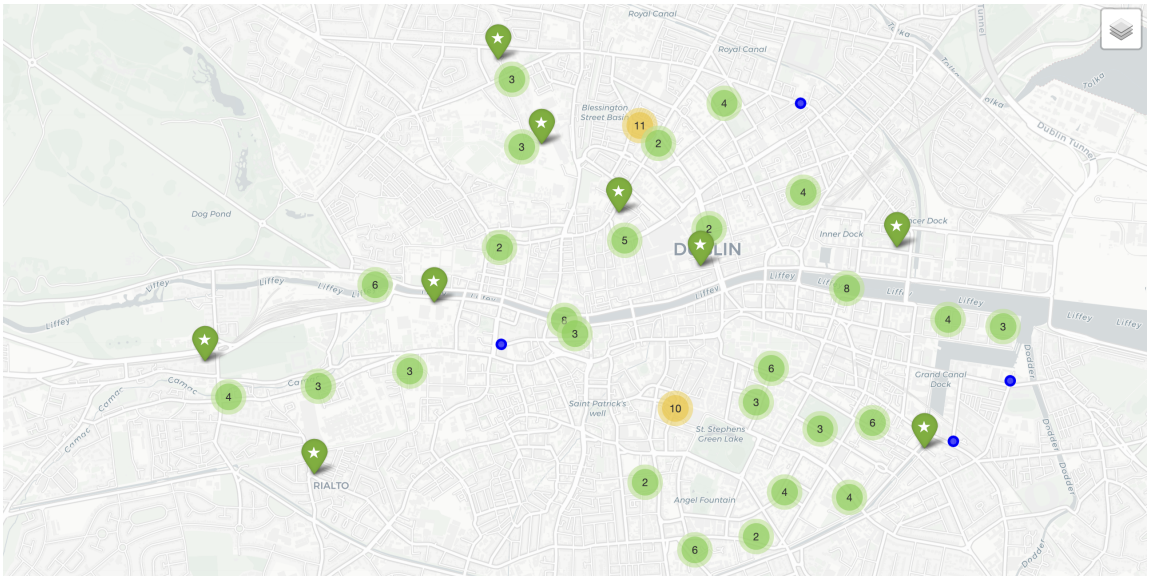

Figure 8: Predicted stations by the GraphSAGE + Focal Loss model (green stars).

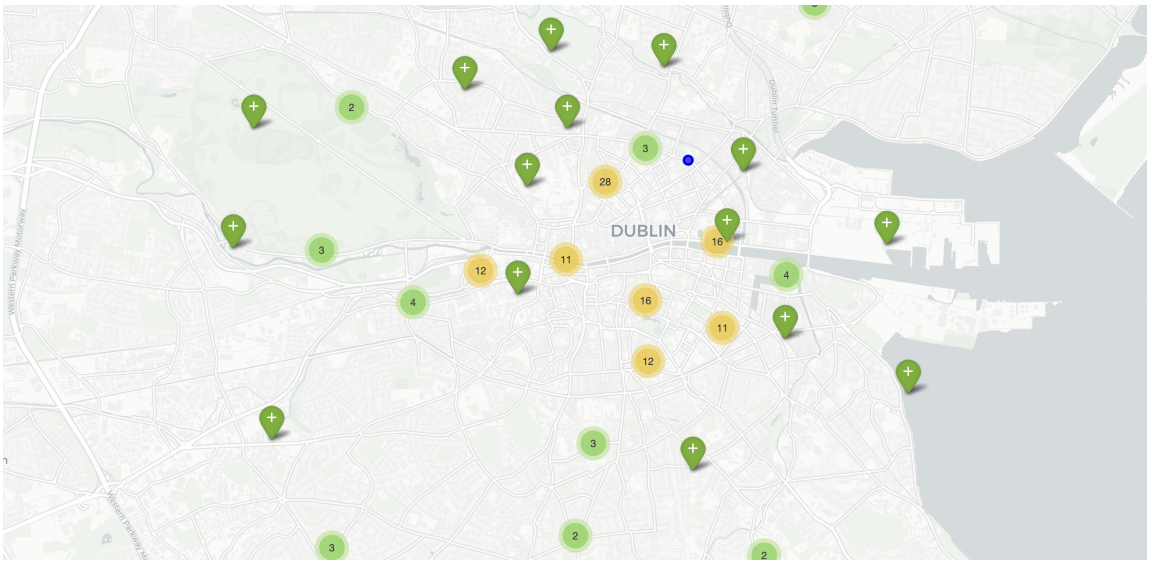

Figure 9: Predicted stations by the GCN + Focal Loss (No Leakage) model (green plus markers).

The GraphSAGE model tends to predict new stations closer to real stations, in particular, in high-demand areas, such as Inner Dock or Grand Canal Dock, this might suggest that the model is more sensitive to usage patterns and sociodemographic features, leading to an efficient expansion.
In contrast, the GCN model with no information leakage model predicts new stations in a more dispersed pattern, including under-served areas zones such as Phoenix Park and suburbs. This may suggest that the model detect spatial gaps in the network, might be influenced by the disconnection of test nodes, which will generalizes better.

6.4 Discussion

Some models performed well and other poorly, in general, the difference is link with the imbalance classes that was develop through experiments. Initially, the initial dataset contains 115 stations, but to make the problem a classification problem I generated the synthetic stations using jittering, resulting in other 115 negative sample (synthetic). However, for the experiments with the GCN with no balance and GraphSAGE, GCN and GAT with focal loss, I expanded the negative class to a proportion of 2:1, by generating 2 synthetic candidates per real station. For instance, the GCN model with no balance the imbalance issue was no address, which lead the model to benefit the majority class and that was the reason for the poor performance.
In contrast, for both models GraphSAGE and GCN models with focal loss the performance improved significantly because focal loss helps the misclassified class. In the case of GCN with focal loss and no leakage information, the model was benefit with addressing the class imbalance and disconnecting the test nodes so that the model does not learn from testing set. For instance, Lin et al. (2017) include focal loss is their study to address the foreground-background imbalance that has in dense object detection. In addition, Zhu et al. (2023) included in their study the disconnection of test nodes from the graph, they point out that if test nodes are connected during training, the model can learn from that information and generate information leakage, inflating the evaluation metrics.

The combination of clustering and the GNN variations provides valuable insights to improve Dublin's bike-sharing network expansion. As Lee and Leung (2023) explained in their study, clustering techniques applied to bike stations and demand patterns can show groups with different profiles and this might help with the selection of potential new locations by focusing in areas similar to the cluster with the highest demand. The clustering analysis allowed us to identify three different groups based on usage and sociodemographic characteristics. For instance, cluster 0 consisted of high-demand stations, in contrast, cluster 1 has underutilized stations or areas with lower usage, indication possible oversupply and cluster 2 represented moderate used stations that might be located in residential zones. These profiles helped guide the selection of new stations by focusing on areas similar to cluster 0.
Furthermore, visualizations indicated that the model did not predict new stations in underserved areas. This is support by the study made by Mohseni et al. (2025) who explained that GNNs are good to capture spatial patterns but when talking about expansion planning it needs to include suburban zones so that areas with low demand are not ignored.

The integration of clustering and those variations support the idea of combining those techniques to predict optimal new stations, this approach addresses efficiency by ensuring availability in important locations and provides a deep and relevant data-driven foundation for expansion planning, helping urban mobility needs, likewise, Roantree et al. (2024) demonstrate that graph-based algorithms can improve the optimization of the dockless bike-sharing network. In general, the approach has the potential to guide smarter urban mobility planning and more equitable access to shared transportation systems.

7 Conclusion

This study aimed to enhance the expansion of Dublin's bike-sharing services by combining clustering techniques and some variations of Graph Neural Network (GNN).
Clustering support decision-making in the bike-sharing services, for instance, after using KMeans clustering to identify three different clusters in the system, these data show how demand patterns can be used to inform strategic planning and understand underutilized areas. Specially, cluster 0 represented high-demand areas with the highest bike availability and usage rates.

In addition, sociodemographic features improve prediction of station demand, because deep learning models are more capable of predicting and understanding data when they have more robust, clean and enriched information, in this study the models were able to give better predictions after enriching the dataset with sociodemographic features (e.g., education, commuting modes, employment status, age) and they learn from the context of each area in Dublin, moreover, the results show that GNNs are effective when learning from complex patterns. For instance, the best performing architecture was GCN combined with focal loss and ensuring no information leakage, achieving an F1-score of 0.976, demonstrating that generalization improve for predicting the best new locations.

Finally, this study aimed to have a more accessible and convenient network, as predicting optimal locations for new bike stations might reduce the distance users need to walk to find a station or find the closest stations with available docks to park, this will result in less frustrations and shorter walk and wait times for the user. In addition, clustering analysis helps the system to detect better underused or oversaturated stations, allowing the system to move bikes from low to high demand zones, and avoiding stations crowding or empty docks during rush times.

7.1 Future Work

Some suggestions can be given for future exploration. First, the dataset could be enriched with additional data, like transportation infrastructure, such as proximity to bus stops, LUAS and DART stations, this will help to capture more connectivity patterns and improve user convenience. Second, the addition of user feedback might show important insights for station performance and perceive service quality. Lastly, the most accurate model, GCN with focal loss and no leakage, can be analyzed in detail to review the potential new locations, this would allow a deep inspection of clusters.

References

Birkins, J. (2023). Comprehensive guide to gnn, gat and gcn: A beginner's introduction to graph neural networks, Medium.

Caggiani, L., Ottomanelli, M. and Murgante, G. (2024). Optimization of bike-sharing repositioning operations: A reactive real-time assignment approach, *Sustainable Futures* **6**: 100154.

Campbell, J. (2023). Bicycle-sharing system.

Chen, Y., Kumar, R. and Sun, M. (2024). Spatio-temporal differences in bike sharing usage: A tale of six cities, *arXiv preprint arXiv:2412.19294* .

Cuong, D. V., Ngo, V. M., Cappellari, P. and Roantree, M. (2024). Analyzing shared bike usage through graph-based spatio-temporal modeling, *IEEE Open Journal of Intelligent Transportation Systems* **5**: 115–131.

GeeksforGeeks (2024). Difference between k-means and dbscan clustering, GeeksforGeeks.

Humranan, P. and Supratid, S. (2023). A study on gcn using focal loss on class-imbalanced bitcoin transaction for anti-money laundering detection, *Proc. Int. Conf. on Computer and Communications (ICCC)*, IEEE.

Lee, C. K. H. and Leung, E. K. H. (2023). Spatiotemporal analysis of bike-share demand using dtw-based clustering and predictive analytics, *Transportation Research Part E: Logistics and Transportation Review* **180**: 103361.

Lei, B., Liu, P., Milojevic-Dupont, N. and Biljecki, F. (2024). Predicting building characteristics at urban scale using graph neural networks and street-level context, *Computers, Environment and Urban Systems* **111**: 102129.

Liang, Y., Ding, F., Huang, G. and Zhao, Z. (2023). Deep trip generation with graph neural networks for bike-sharing system expansion, *Transportation Research Part E: Logistics and Transportation Review* **177**: 103349.

Lin, T. Y., Goyal, P., Girshick, R., He, K. and Dollár, P. (2017). Focal loss for dense object detection, *2017 IEEE International Conference on Computer Vision (ICCV)*, IEEE, Venice, Italy, pp. 2999–3007.

Liu, T., Jiang, A., Zhou, J., Kwan, H. K. et al. (2023). Graphsage-based dynamic spatial–temporal graph convolutional network for traffic prediction, *IEEE Transactions on Intelligent Transportation Systems* **PP**(99): 1–15.

Liu, X., Lu, J., Chen, X., Fong, Y. H. C., Ma, X. and Zhang, F. (2023). Attention based spatio-temporal graph convolutional network with focal loss for crash risk evaluation on urban road traffic network based on multi-source risks, *Accident Analysis and Prevention* **190**: 107262.

Ma, L., Ye, R. and Wang, H. (2021). Exploring the causal effects of bicycling for transportation on mental health, *Transportation Research Part D: Transport and Environment* **93**: 102773.

McCárthaigh, S. (2023). The economic cost of congestion in the greater dublin area 2022–2040, *Technical report*, Strategic Research and Analysis Division.

Mohseni, G., Nourinejad, M. and Park, P. Y. (2025). Bike-sharing ridership prediction for network expansion using graph neural networks, *Technical report*, University of Toronto. Preprint.

Rallabandi, S. (2021). Activation functions: Relu vs leaky relu, Medium.

Roantree, M., Cuong, D. V., Murphy, N. and Ngo, V. M. (2024). Graph-based optimisation of network expansion in a dockless bike sharing system, *Technical report*, Insight Centre for Data Analytics, Dublin City University.

Sokolova, M. and Lapalme, G. (2009). A systematic analysis of performance measures for classification tasks, *Information Processing & Management* **45**(4): 427–437.

Sun, J., He, Y. and Zhang, J. (2023). A cluster-then-route framework for bike rebalancing in free-floating bike-sharing systems, *Sustainability* **15**(22): 15994.

Tang, Z., Wang, J. and Zhang, W. (2022). Improving short-term bike-sharing demand forecast with irconv+lstm, *Transportation Research Part C: Emerging Technologies* **142**: 103798.

Xiao, Y., Ahmed, F. and Sha, Z. (2023). Graph neural network-based design decision support for shared mobility systems, *Journal of Mechanical Design* **145**(1): 1–17.

Xin, R., Yang, J., Ai, B., Ding, L., Li, T. and Zhu, R. (2023). Spatiotemporal analysis of bike mobility chain: A new perspective on mobility pattern discovery in urban bike-sharing system, *Journal of Transport Geography* **109**: 103606.

Zhan, Z., Guo, Y., Noland, R. B., He, S. Y. and Wang, Y. (2023). Analysis of links between dockless bikeshare and metro trips in beijing, *Transportation Research Part A: Policy and Practice* **175**: 103784.

Zhang, H., Wang, Q. and Li, Y. (2023). Spatio-temporal neural structural causal models for bike flow prediction, *Cities* **142**: 104607.

Zhao, T., Huang, Z., Tu, W., He, B., Cao, R., Cao, J. and Li, M. (2024). Coupling graph deep learning and spatial-temporal influence of built environment for short-term bus travel demand prediction, *Computers, Environment and Urban Systems* **101776**.

Zhou, L., Liu, F. and Zhang, M. (2025). Machine learning methods to identify commuter bike-sharing activities, *Journal of Transport Geography* **115**: 103603.

Zhu, J., Zhou, Y., Ioannidis, V. N., Qian, S., Ai, W., Song, X. and Koutra, D. (2023). Pitfalls in link prediction with graph neural networks: Understanding the impact of target-link inclusion and better practices.