Route Recommendation Based on POIs and Public Transportation

Ágata Palma[1][a], Pedro Morais[1][b] and Ana Alves[1,2][c]

[1]*Polytechnic University of Coimbra, Rua da Misericórdia, Lagar dos Cortiços, S. Martinho do Bispo, 3045-093 Coimbra, Portugal*

[2]*CISUC, LASI, University of Coimbra, Polo II, Pinhal de Marrocos, 3030-290, Coimbra, Portugal*

{a2023113935, a21280686, aalves}@isec.pt

Keywords: GIS, Information Retrieval, Ambient Intelligence, Clustering, Route Recommendation, POIs.

Abstract: With the rapid advancement of technology in today's interconnected world, Ambient Intelligence (AmI) emerges as a powerful tool that revolutionizes how we interact with our environments. This article delves into the integration of AmI principles, Python programming, and Geographic Information Systems (GIS) to develop intelligent route recommendation systems for urban exploration. The motivation behind this study lies in the potential of AmI to address challenges in urban navigation, personalized recommendations, and sustainable transportation solutions. The objectives include optimizing travel routes, promoting sustainable transportation options, and enhancing user experiences. This research will contribute to advancing AmI technologies and their practical applications in improving urban living standards and mobility solutions.

1 INTRODUCTION

Ambient Intelligence (AmI) represents a new paradigm in computing that aims to embed intelligence into everyday environments. It involves the integration of computational capabilities into ordinary objects, allowing them to interact with users and each other in a natural and intelligent manner. AmI emphasizes the presence of humans alongside smart interfaces that can adapt to human emotions, behaviors, and expectations. This concept envisions the creation of smart environments, such as smart homes, smart healthcare facilities, and smart cities, where everyday objects are seamlessly connected and capable of enhancing daily living experiences. AmI is seen as a significant societal and cultural shift, with the potential to transform the way people interact with technology and their surroundings (Thankachan, 2023).

Since AmI takes advantage of sensors and Internet of Things (IoT) devices to gather information about the surrounding environment, it is a useful tool to make inferences based on proximity, intent, and behavioral patterns. This facilitates personalized experiences, as for example, receiving location-based alerts when reaching points of interest (POIs) in a new city(Mahmood et al., 2023).

Ambient Intelligence enhances the accuracy and relevance of environmental data by incorporating Geographic Information Systems (GIS) and Information Retrieval techniques. GIS offers spatial analysis and mapping to understand user interactions geographically, while Information Retrieval efficiently extracts relevant data for context-aware recommendations. Clustering techniques group similar data points to identify patterns in user behavior, aiding route recommendation systems by predicting optimal paths based on historical data and preferences. These technologies enable AmI to create intelligent environments that anticipate and respond to user needs, providing seamless and enriched interactions.

The motivation behind this study lies in the practical application of advanced geospatial technologies and algorithms to enhance urban navigation. The aim is to develop an intelligent system that can provide efficient, customizable routing solutions tailored to individual preferences, particularly in urban environments where efficient navigation and personalized experiences are crucial to navigate busy zones and discover POIs. Therefore, these intelligent systems can offer context-aware recommendations and optimize routes tailored to user preferences and sustainable transport options, ultimately promoting efficient travel and contributing to sustainable urban mobility.

[a] https://orcid.org/0009-0009-4450-700X
[b] https://orcid.org/0009-0003-5962-0386
[c] https://orcid.org/0000-0002-3692-338X

Palma, Á., Morais, P. and Alves, A.
Route Recommendation Based on POIs and Public Transportation.
DOI: 10.5220/0013006800003838

In *Proceedings of the 16th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management (IC3K 2024) - Volume 1: KDIR*, pages 391-398
ISBN: 978-989-758-716-0; ISSN: 2184-3228

This work explores the integration of AmI principles, Python programming, and GIS to develop intelligent route recommendation systems for urban exploration based on POIs and available public transportation. Given a city and, optionally, a category, the system will reply with a list of POIs and a suggested route to visit the largest number of POIs in the shortest route possible using public transportation. The base of this work is a variant of the Travelling Salesman Problem (TSP), which involves finding the shortest possible route that visits a set of given locations exactly once and then returns to the starting point (Özcan and Kaya, 2018). The challenge in this research is similar to the one on TSP: determine the most efficient route between multiple POIs while minimizing the total travel distance.

The structure of the article includes a review of related works and AmI principles and their relevance in urban navigation, followed by a discussion of technical aspects such as data integration, route optimization algorithms, and visualization techniques using platforms like Quantum GIS (QGIS).

Practical implications, potential extensions, and the broader impact of AmI-driven solutions on urban mobility and city planning are also addressed in the discussion and conclusions sections. This work aims to contribute to the advancement of technology that enhances user experiences and promotes sustainable and efficient mobility solutions in urban settings.

2 RELATED WORK

In this section, a brief literature review is presented, focusing on existing applications, systems, and studies that share similar objectives or themes related to AmI and intelligent route recommendation systems for urban exploration based on POIs.

Based on the TSP, (Özcan and Kaya, 2018) aimed to create a new tourist guide app using OpenStreetMap (OSM). To achieve this, the study involved various tasks using OSM tools, libraries, and frameworks. These tasks included real-time area drawing on OSM, path computation, selection of POIs, and map understanding. The app intends to determine the shortest route between user-selected destinations, optimizing travel time and displaying the route visually on the map. The Hill Climbing Algorithm (HCA), known for its memory efficiency and local search approach, was used for the TSP.

On the itinerary recommendation variant, (Panagiotakis et al., 2022) proposed a method to personalize itinerary recommendation (PIR) with POIs categories, for tourists tours. The authors' method was based on the Expectation Maximization (EM) algorithm, and solves, sequentially, the PIR problem by selecting POIs that maximize a suitable objective function, such as user satisfaction, user time budget, POIs opening hours, POIs category and spatial constraints. In a similar scope, (Lou, 2022) focused on categorizing POIs but with an improved k-means algorithm to be applied to intelligent tourism route planning. The proposed scheme considers tourists' preferences and aims to find the shortest route between desired locations within a selected area.

(Mahdi et al., 2023) also redirected their research focus towards POIs. They applied regression models to analyze the data obtained from Google Popular Times (GPT) to predict the amount of time people would spend at POIs. With this contribution, a similar process would be possible to improve the route generation plan when time constraints are a variable.

Besides the prediction of the time spent at a POI, when planning a route based on public transportation, it is also crucial to take into account the time spent from one point to another. (Zhang et al., 2022) state the importance of improving travel time prediction. The study highlights the importance of real-time, accurate, reliable and low-cost multi-source data for better predictions. The authors affirm that the traditional methods for predicting travel time are deficient and a new approach based on intelligent technology would improve the prediction accuracy. In order to accomplish this, a prediction model based on the Kalman filter - high accuracy in one-step prediction - was designed. For this model, two sub-modules were created: the Route Travel Time Prediction Model - predicts travel time for an entire bus route - and the Stop Dwell Time Prediction - predicts the time spent at bus stops. In this study, the data sources used included GPS (Global Positioning System), AFC (Automatic Fare Collection), and IC (Integrated Circuit) and the models were validated using Automatic Vehicle Location (AVL) from real world scenarios. The results indicate the prediction model meets accuracy requirements for travel time prediction.

(Sarridis et al., 2022) proposed a personalized route recommendation system that balances the trade-off between distance and POIS using hypergraph models. Their framework considers tourist satisfaction and leverages both visual and geographical data to optimize the shortest path algorithm through POI images embedded in a hypergraph. Similarly, (Karantaidis et al., 2021) applied multi-stage optimization learning in hypergraph structures for image and tag recommendations, dynamically updating hypergraph structures and hyperedge weights to achieve higher accuracy in POI ranking and recommendations.

The contribution of (Li et al., 2021) to this work is based on a solution for another problem. Instead of the common questions such as "find the k nearest POIs around me" or "give me the bus plan from *s* to *d*", the authors proposed a method to answer the "give me *k* POIs that I can reach earliest within one transfer by bus". In the public transportation network (PTN), the users' primary concern is "which POI can be reached with the least travel time under some specified transfer numbers considering the different departure time and frequency of buses". To answer the proposed question, the k-nearest neighbor (kNN) query should be applied. Given a set of information—POIs, a PTN, a location, departure time, and a transfer number constraint—the kNN query returns the k POIs that meet these conditions.

Another possible method for location-based systems, besides kNN, is the Multi-Cost Transportation Network-constrained skyline query (MCTN-CSQ). (Gong et al., 2020) implemented the CSQ System, the first of its kind, as a web application supporting constrained skyline query on multi-cost transportation networks. Users input query points and receive skyline answer-objects reachable via transportation networks, superior on at least one dimension. For example, lets assume a user needs to book a room for the night but he has more constraints about the desired room: it can be reached by taking public transportation, and the transportation fare and the travel time should be reasonable - the query processing component of the CSQ System handles the query execution. "The system is implemented as a web application, which allows users to input a query point from a web interface, get the skyline result by using several algorithms, and display the result on the web interface" (Gong et al., 2020).

3 SYSTEM ARCHITECTURE

The application is designed to provide a comprehensive solution for route planning and analysis within the QGIS environment. The system architecture is composed of several elements, interconnected to facilitate preprocessing, route generation, spatial analysis, visualization, and user interaction.

The user interface, implemented using the QGIS interface in the first phase and a plugin in QGIS in the second phase, serves as the entry point for users to select the city and visualize the suggested route, as well as to specify POI categories. The route generation engine employs an algorithm to compute the optimal route, connecting different POIs based on user-defined parameters. The aim is to find the shortest path in the road network integrating public transport routes and stops.

In the first phase, a proof of concept is achieved by working with the available processing tools and plugins in QGIS, such as ORS (OpenRouteService) tools. The system uses data provided for the development of this project, specifically POIs and roads in Portugal and public transportation in Coimbra. The spatial visualization is handled by the QGIS environment, taking advantage of HeatMaps and route overlays to visually present analysis results, as well as to produce a georeferenced PDF.

For the second phase, a custom plugin is developed to provide the user with a more friendly and intuitive interface. Using the user's input for a location and category of POIs, and the processing of POIs with machine learning methods, a route is drawn using the shortest path possible across the region with the most of these POIs.

4 DATA SOURCES

To populate the application with pertinent data concerning POIs, the primary source relies on OpenStreetMap for the second step, retrieved through the "osmnx" python package.

For the first step, data containing POIs and roads of Portugal in shapefiles format were provided in the context of this project, as well as public transportation data for Coimbra, with routes and stops for SMTUC (Serviços Municipalizados de Transportes Urbanos de Coimbra). Additionally, the application integrates (1) the QGIS plugin QuickOSM to retrieve the boundaries of Coimbra city and (2) the module Quick Map Services (QMS) to procure a standardized raster layer of OSM.

5 MACHINE LEARNING

To enhance the performance of the application, Clustering is applied, which allows the identification of groups of POIs that are geographically close to each other. Through this unsupervised learning method, the most concentrated area of POIs is identified and a route is established within that zone. A density-based cluster analysis algorithm, the Density-Based Spatial Clustering of Applications with Noise (DBSCAN), is applied due to its robustness and effectiveness in handling spatial data. As noted in (Lou, 2022), DBSCAN has the great advantage of clustering dense datasets of any shape and is "sensitive to the selection of initial values, but insensitive to noise points and has certain noise immunity". Another advantage is the unnecessary need to predefine the number of clusters. The initial DBSCAN parameters are:

- **eps** (ε). The maximum distance between two points to be considered as part of the same neighborhood. This parameter defines the radius of the neighborhood around each point.
- **minPts.** The minimum number of points required to form a dense region. A point is considered a core point if it has at least *minPts* within its *eps* radius.

These parameters are crucial for the performance of the DBSCAN algorithm. In this study, *eps* and *minPts* are fine-tuned based on the spatial distribution of POIs in the dataset.

The clustering process entails loading the road network graph using *OSMnx*, projecting the POIs to align with the Coordinate Reference System (CRS) of the graph, and generating a distance matrix based on network distances. Subsequently, the DBSCAN algorithm is employed to detect clusters, and the outcomes are assessed using metrics like Silhouette Score and Davies-Bouldin Index.

The Silhouette Score measures how similar a point is to its own cluster compared to other clusters. Higher values indicate well-defined clusters with clear separation between them. The Davies-Bouldin Index assesses the average similarity ratio of each cluster with its most similar cluster, where lower values indicate better-defined clusters with less overlap. These metrics provide a quantitative evaluation of the clustering quality, ensuring that the clusters formed are meaningful and accurate.

6 VISUALISATION OF DATA

Throughout the first and second phases, different approaches are utilized for collecting, treating, and displaying the results. Both approaches are addressed, demonstrating the evolution and refinement of the methods to achieve a more automated response in custom route generation.

6.1 Phase I

Taking advantage of the already present module in QGIS, QMS, the standardized raster layer is retrieved, providing a comprehensive and detailed map background in *EPSG:4326*. This CRS, also known as *WGS 84*, is widely used in geographic coordinate systems and is the one that the ORS API expects in the requests.

To commence data analysis, the shapefiles of Portugal's POIs and roads are imported, along with the SMTUC General Transit Feed Specification (GTFS) containing route information and bus stops, facilitated by the GTFS GO plugin. Furthermore, the polygon delineating the region of Coimbra is imported using the QuickOSM plugin. Additionally, a new polygon is drawn within the Coimbra region to delimit the analysis area.

A new layer, named *Coimbra_POIS* is created through the extraction by location of elements that intersect or are contained within the area of the polygon. This layer is subsequently utilized as the foundation for generating a HeatMap, providing a visual representation of the concentration of POIs within the delimited area. Since meters are preferred over degrees for measurements, the layers are re-projected to *EPSG:3763*. This adjustment enables the proper configuration of parameters for the DBSCAN algorithm (ε: 200 meters; minPts: 4), using the *Coimbra_POIS* layer as the data source. The outcomes demonstrate a clear separation of clusters, indicating a satisfactory fit, as shown in Figure 1. To enhance visualization and delineate cluster regions more distinctly, concave hulls are employed for each cluster. A concave hull is a shape that closely wraps a set of points, capturing the boundaries of the points more accurately. The result provided a collection of polygons encompassing the points within each cluster, as depicted in Figure 2.

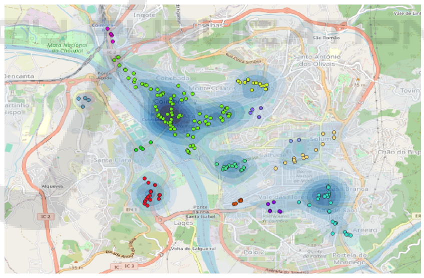

Figure 1: Heatmap with DBSCAN clustered POIs.

In this phase, the simulation involves a user who wishes to travel from point A to point B, as depicted in Figure 2, utilizing the shortest path and public transportation services. With this goal in mind, a manual approach is adopted for route construction. Using the ORS tools, a few points are manually selected as coordinates to create two custom routes, employing the shortest path and driving-car preferences. After re-projecting both custom and SMTUC routes and stops, each route is segmented into sections of approximately 500 meters, resulting in the creation of two new layers: (1) SMTUC sections intersecting the custom routes and (2) SMTUC stops along the custom route. Additionally, leveraging ORS tools, a new layer with isochrones is created, as seen in Figure 3. An isochrone is a line or boundary on a map that connects points representing equal travel time or distance from a particular location. This new layer provided a visual analysis of the area accessible from each SMTUC stop along the route within 2, 5, and 7-minute thresholds.

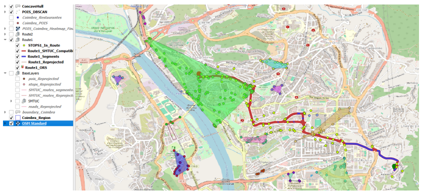

Figure 2: Route 1.

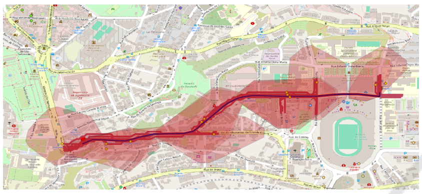

Figure 3: Route 2 with isochrones for 2 minutes.

Given the limitations of the ORS API for isochrones, no route with them is created to the center of the largest cluster of POIs. Nonetheless, through visual analysis, it can be confirmed that using this approach could indeed provide an effective tool for route customization based on POI concentration and public transportation.

6.2 Phase II

In the second phase, the system allows users to select POIs from any geographic location. These POIs are organized into top-level categories, each containing subcategories. For example, the Tourism category includes Hotels and Museums, the Amenity category features Bars and Cafes, and the Shop category encompasses Malls.

The main objective of this stage is to provide a more automated response to the challenge presented in the first phase of this project. A plugin for QGIS has been developed - Optimal Custom Route - which offers an intuitive and efficient tool for route planning and visualization.

The development environment includes *OSMnx* for geographic data handling, *OpenRouteService* for routing, *gpxpy* for GPS Exchange Format (GPX) file manipulation, *scikit-learn* for clustering, and *geopy* for geocoding. These libraries provide the necessary tools for implementing the plugin's core functionalities and can be installed using the following commands:

```
$pip install osmnx
$pip install openrouteservice
$pip install gpxpy
$pip install scikit-learn
$pip install geopy
```

The next step focuses on designing and implementing the user interface, developed using PyQt5. This interface comprises two main windows:

1. The first window allows users to input the name of the city for which they want to plan a route.
2. The second window is dedicated to route customization details, as shown in Figure 4.

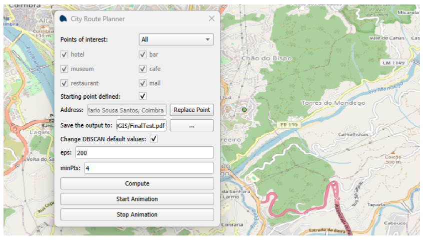

Figure 4: Customization window with a starting point defined and customized DBSCAN parameters.

To handle geographic data, the plugin utilizes "OSMnx" to collect and process map data from OpenStreetMap. The process commences with geocoding the city name to acquire geographic coordinates. These coordinates are subsequently utilized to import the relevant map tiles into QGIS as layers, thereby offering users a visual representation of the area of interest.

For selecting POIs, users can choose from various categories, such as tourism, amenities, and shops. The plugin dynamically generates checkboxes for each subcategory, allowing for detailed selection. Once the POIs are selected, the plugin retrieves the corresponding data from OpenStreetMap and saves the data in a new layer with the name, category, and subcategory of the POI. Then, it proceeds to clustering using the pre-defined values of $\varepsilon = 200$ meters and minPts = 4 or custom values chosen by the user.

Clustering analysis plays a crucial role in the plugin, focusing on grouping POIs based on geographic proximity. To use the collected data, a transformation is needed. In this phase, the coordinates of the POIs are converted to a suitable coordinate system to ensure accurate distance measurements. Typically, the Universal Transverse Mercator (UTM) projection is used because it provides a more accurate representation of distances compared to latitude and longitude. This is essential for spatial data analysis, as the DBSCAN algorithm operates on distances between points. To ensure consistency in distance calculations, the POIs data is projected into the same CRS as the road network graph generated by *OSMnx*. This CRS transformation is important for aligning the POIs with the graph, allowing for accurate integration and subsequent analysis.

Once the data is transformed, a road network graph is created using *OSMnx*. This graph represents the road network within the specified place, where nodes correspond to intersections, and edges represent road segments connecting these intersections. The road network graph is truncated to retain only the largest connected component. This step is needed to avoid isolated nodes that do not contribute to the main network, ensuring a coherent and comprehensive road network for analysis. The truncated graph provides a strong foundation for mapping POIs and calculating network distances.

With the road network graph prepared, the POIs are projected into the same CRS as the graph to maintain consistency, and the nearest nodes in the road network graph are found for each POI. This way, distances between POIs can be calculated within the context of the road network.

The clustering process involves calculating a distance matrix, which is essential for applying the DBSCAN algorithm. Initially, a pairwise Euclidean distance matrix is calculated between the nodes representing the POIs. However, for more accurate distance measurements that account for the road network, this Euclidean distance matrix is converted into a network distance matrix using Dijkstra's algorithm. This algorithm computes the shortest path between nodes based on the actual road network distances. By using the network distance matrix, the DBSCAN algorithm can accurately cluster POIs based on real-world distances, rather than straight-line distances. The DBSCAN algorithm is then applied to this network distance matrix. The eps parameter, which is used in meters, defines the maximum distance between two points for them to be considered part of the same cluster. The minPts parameter specifies the minimum number of points required to form a dense region. The metric *precomputed* is used to indicate that the distance matrix has already been calculated.

After the clustering is performed, the results are processed to extract meaningful clusters. Noise points, which are points labeled as -1 by DBSCAN, are excluded from further analysis and the largest and densest clusters are identified. Subsequently, the outcomes are assessed using Silhouette Score and Davies-Bouldin Index.

These clusters are then visualized within the QGIS environment, providing an intuitive and comprehensive view of the spatial distribution of POIs. This visualization aids in identifying key areas of interest, as shown in Figure 5 and supports the generation of an optimal route.

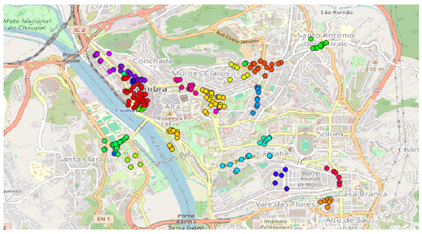

Figure 5: Clustered POIs.

Based on the clustered POIs, the route generation process initiates a request to the ORS API to compute the optimal route. Users can specify the starting point either manually or by utilizing the center of the largest cluster (the default by omission). Subsequently, the plugin selects waypoints from the largest cluster, applies a greedy TSP solver to determine the optimal order of waypoints, and generates the route using ORS. The resulting route is then visualized in QGIS (see Figure 6), providing users with an interactive map display. To enhance the user experience, the plugin includes a route animation feature, which reads GPX data and animates the movement along the route on the QGIS map canvas. Finally, it also supports exporting the created route and layers to a PDF, as an image.

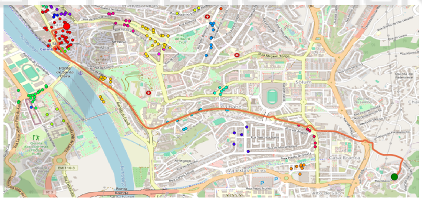

Figure 6: Route visualization in QGIS.

The integration of clustering analysis and advanced route generation techniques in the second phase represents a significant advancement in developing intelligent route recommendation systems. By allowing users to select POIs from a variety of categories and subcategories, the system provides highly personalized and efficient routing solutions. The use of DBSCAN for clustering POIs based on real-world distances ensures accurate and meaningful groupings, while the ORS API facilitates the generation of optimized routes. The inclusion of a user-friendly interface, interactive map displays, and features such as route animation and export options enhances the overall user experience, making the system a powerful tool for urban exploration and navigation.

7 EVALUATION

To assess the success of this work, the system is evaluated on functionality, user experience, performance, clustering effectiveness, and accuracy. The system's ability to accurately recommend routes based on user-selected cities and POI categories is assessed, as well as the effectiveness of the route generation engine in optimizing factors like distance (for phase II) and available public transportation options (for phase I), in real-time route recommendations. Each project phase meets expectations, demonstrating flexibility and efficiency in using user inputs to recommend routes within concentrated areas of interest. This aligns with the motivations described by (Mahmood et al., 2023), who highlights the importance of context-aware recommendations and optimized routes tailored to user preferences.

Regarding performance evaluation, the system demonstrates high efficiency under varying loads. In Phase I, a significant number of POIs are retrieved without delay. In Phase II, although fewer POIs are processed, more complex operations are executed in sequence (API calls followed by clustering analysis, display, and data export) within a few seconds. This real-time response capability underscores the practical application of advanced geospatial technologies and algorithms in urban navigation, as discussed in the introduction.

The reliability of clustering effectiveness in identifying concentrated areas of POIs and generating optimized routes within those zones is also assessed. To ensure clustering effectiveness, two metrics are used for validation: the Silhouette Score and the Davies-Bouldin Index. These metrics provide quantitative evaluations of clustering quality, confirming that the system effectively identifies meaningful clusters of POIs. However, some challenges are encountered in clustering effectiveness in phase II, particularly with results suggesting an overlap of clusters when using the same parameters as in phase I. This can be visualized in the layers and in the Silhouette Score with values ranging from -0.7 to -0.4 and the Davies-Bouldin Index with values from 1 to 2 or 3, depending on the parameter values. This cluster overlap could be attributed to the presence of various sources and categories for the POIs. In the first step, all the retrieved POIs are used for the clustering process, and in the second step, only a few categories are processed. Nonetheless, the overall results are promising. This finding corroborates the work of (Lou, 2022), who emphasizes the importance of accurate clustering for intelligent tourism route planning.

The system's clustering process benefits from the use of the DBSCAN algorithm, known for its robustness in handling spatial data and noise, as noted by (Lou, 2022). The application of DBSCAN, along with the conversion of Euclidean distance matrices into network distance matrices using Dijkstra's algorithm, allows for accurate clustering based on real-world distances. This approach is consistent with the findings of (Zhang et al., 2022), who highlights the importance of accurate distance measurements and intelligent technology in improving travel time predictions and route optimization.

In terms of user experience, the development of a custom QGIS plugin [1],"Optimal Custom Route", provides an intuitive and efficient tool for route planning and visualization. The user interface, designed using PyQt5, offers a seamless and interactive experience for selecting POIs and generating routes. The integration of clustering analysis, route optimization, and visualization within the QGIS environment enhances the system's usability and practicality, aligning with the envisioned AmI principles of creating smart environments that enhance daily living experiences (Thankachan, 2023).

In conclusion, the application achieves its primary goals of generating optimal routes that connect different POIs within selected cities, demonstrating high accuracy in visualization and spatial analysis results. By identifying areas with high concentrations of POIs (in both phases) and public transportation coverage (in Phase I), the system successfully provides personalized and efficient navigation solutions. Future work will focus on enhancing clustering techniques, integrating real-time data, optimizing performance, incorporating user feedback, and expanding the range of POIs categories to further improve the system's functionality and applicability.

8 CONCLUSIONS

This study successfully integrates AmI principles, Python programming, and GIS to develop intelligent route recommendation systems for urban exploration. The developed systems optimize travel routes, promote sustainable transportation options, and enhance user experiences. The research demonstrates the potential of AmI to address challenges in urban navigation and personalized recommendations, contributing to sustainable urban mobility.

The development process is divided into two phases. In the first phase, existing QGIS tools and plugins are utilized to manually create and ana-

[1] https://github.com/AgataPalma/OptimalCustomRoute

lyze routes based on POIs and public transportation data. This phase demonstrates the feasibility of using geospatial technologies to optimize urban navigation. The second phase involves the creation of a custom QGIS plugin, "Optimal Custom Route," providing an automated and user-friendly interface for route planning and visualization. This phase leverages advanced machine learning techniques, specifically DBSCAN clustering, to identify dense areas of POIs and generate optimized routes.

The system's performance is evaluated based on functionality, user experience, performance, clustering effectiveness, and accuracy. The results indicate that the system accurately recommends routes based on user-selected cities and POI categories, efficiently handles varying loads, and generates well-defined clusters of POIs. However, some challenges are encountered, particularly in clustering effectiveness when dealing with different sources and categories of POIs, which will need further refinement.

8.1 Future Work

While the current system shows promising results, several areas for future work can enhance its functionality and applicability:

- **Enhanced Clustering Techniques.** Future research could explore more advanced clustering algorithms and parameter tuning to improve clustering effectiveness, particularly when dealing with diverse categories of POIs.
- **Integration with Real-time Data.** Incorporating real-time data from public transportation systems, traffic conditions, and user location can enhance the system's ability to provide dynamic and real-time route recommendations.
- **Extended POI Categories:** Expanding the range of POI categories and integrating additional data sources can provide more comprehensive and personalized route recommendations.
- **Mobile Application Development.** Developing a mobile app of the system can make it more accessible to users on the go, providing seamless and interactive route recommendations.
- **Sustainability Metrics.** Incorporating sustainability metrics, such as carbon footprint reduction and energy efficiency, into the route optimization process can further promote sustainable urban mobility solutions.

In conclusion, this research demonstrates the significant potential of integrating AmI, Python programming, and GIS in developing intelligent route recommendation systems. By addressing the identified challenges and exploring future research directions, ongoing advancements in AmI technologies and their practical applications can continue to improve urban living standards and mobility solutions.

REFERENCES

Gong, Q., Liu, J., and Cao, H. (2020). Csq system: A system to support constrained skyline queries on transportation networks. In *Proc. of Intl. Conf. on Data Engineering*, volume 2020-April, pages 1746–1749. IEEE Computer Society.

Karantaidis, G., Sarridis, I., and Kotropoulos, C. (2021). Adaptive hypergraph learning with multi-stage optimizations for image and tag recommendation. *Signal Processing: Image Communication*, 97.

Li, J., Zhang, L., Ni, C., An, Y., Zong, C., and Zhang, A. (2021). Efficient k nearest neighbor query processing on public transportation network. In *Proc. of 20th Intl. Conf. on Trust, Security and Privacy in Computing and Communications, TrustCom*, pages 1108–1115. IEEE.

Lou, N. (2022). Analysis of the intelligent tourism route planning scheme based on the cluster analysis algorithm. *Computational Intelligence and Neuroscience*, 2022:3310676.

Mahdi, A. J., Tettamanti, T., and Esztergar-Kiss, D. (2023). Modeling the time spent at points of interest based on google popular times. *IEEE Access*, 11:88946–88959.

Mahmood, M. R., Kaur, H., Kaur, M., Raja, R., and Khan, I. A. (2023). *Ambient intelligence and internet of things: An overview*, chapter 1, pages 1–32. John Wiley & Sons, Inc., 1 edition.

Panagiotakis, C., Daskalaki, E., Papadakis, H., and Fragopoulou, P. (2022). The tourist trip design problem with poi categories via an expectation-maximization based method.

Sarridis, I., Karantaidis, G., and Kotropoulos, C. (2022). Image driven optimal personalized route recommendation. In *IEEE IVMSP 2022*. Institute of Electrical and Electronics Engineers Inc.

Thankachan, D. (2023). *Introduction to Ami and IoT*, chapter 12, pages 361–381. John Wiley & Sons, Inc., 1 edition.

Zhang, X., Lauber, L., Liu, H., Shi, J., Xie, M., and Pan, Y. (2022). Travel time prediction of urban public transportation based on detection of single routes. *PLoS ONE*, 17:e0262535.

Özcan, S. and Kaya, H. (2018). An analysis of travelling salesman problem utilizing hill climbing algorithm for a smart city touristic search on openstreetmap (osm). In *2nd Intl Symposium on Multidisciplinary Studies and Innovative Technologies (ISMSIT)*, pages 1–5.