UCLouvain

epl

École polytechnique de Louvain

Public Transport Grid Analysis Using Multi-scale Backbones

Auteur: **Breno TIBURCIO**
Promoteur: **Jean Charles DELVENE**
Lecteurs: **Julien HENDRICKX, Laurent JACQUES**
Année académique 2023-2024
Master [120] : ingénieur civil en sciences des données

Declaration of Authorship

I, BRENO SOARES TIBURCIO, declare that this thesis titled, 'Public Transport Grid Analysis using Multi-scale Backbones' and the work presented in it are my own. I confirm that:

- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- Where I would like to acknowledge the assistance provided by ChatGPT, DeepL, and Grammarly in proofreading, grammar correction, translation, and refinement of this thesis.
- I have acknowledged all main sources of help.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Signed: Breno Soares Tiburcio

Date: 03/06/2024

"People who want to understand democracy should spend less time in the library with Aristotle and more time in the buses and subways."

Simeon Strunsky

UNIVERSITÉ CATHOLIQUE DE LOUVAIN

Abstract

EPL - École Polytechnique de Louvain
INMA - Département d'ingénierie mathématique

Master of Sciences

by Breno Tiburcio

By modeling public transportation systems and experimenting with a few scenarios coupled with graph theory methods, this study aims to discuss their approaches while also examining infrastructural, operational guidelines, and real implementation challenges. The realm of graph theory faces no limitations, as nodes and edges can be freely created. However, a transportation grid must respect urban spaces and topological constraints such as leveling off and lakes, for example. Still, analytical tools can help us identify operational opportunities, map expansion, and optimize efforts to bridge delivery and demand. In this study, we will present different rankings to highlight the relevance discrepancy between infrastructure, service demand, and potential stress (according to toy metrics). We will also discuss other possible metrics to help identify service quality gaps between areas, such as a mobility index based on the number of stations, vehicle speed, and average speed. In other words, it sheds light on the fact that some neighborhoods are better served by the transport system, raising important moral questions about profitability and social responsibility through a common service. Additionally, we extrapolate on how machine learning can be used to bring efficiency without affecting the equality of our common right to come and go.

Acknowledgements

I extend my sincere gratitude to my thesis advisor, Jean Charles Delvenne, for his invaluable guidance and support throughout this project.

To my parents, Rosana and Mauricio, and my siblings, Gustavo, Gabriela, and Camila, your constant encouragement despite the distance has been a source of strength.

Special thanks to my classmates for their collaboration on various projects and the friendships we've forged during the Master's course. Your camaraderie and shared experiences have enriched my learning journey.

I appreciate Transport for London for providing access to the data used in this research, and Université Catholique de Louvain for the conducive academic environment.

Thank you to everyone who contributed to making this project a success.

Contents

List of Figures

List of Tables

Abbreviations

NaPTAN	**Na**tional **P**ublic **T**ransport **A**ccess **N**odes
MTT	**M**onday **T**o **T**hursday
FRI	**FRI**day
SAT	**SAT**urday
SUN	**SUN**day
NUMBAT	**NUMB**er **AT**
TFL	**T**ransport **F**or **L**ondon
DLR	**D**ocklands **L**ight **R**ailway
OG	**O**ver**G**round
ALP	**ALP**ha Method
API	**A**pplication **P**rogram **I**nterfaces
THR	**THR**esholding Method
JSON	**J**ava**S**cript **O**bject **N**otation

Symbols

s_i	node strength	unitary
k_i	node degree	unitary
$\omega_{i,j}$	edge weight	unitary
$\rho(\omega, k_i)$	weight prob. density function	dimensionless
$\alpha_{i,j}$	edge statistical relevance	dimensionless

To My Beloved Family and Dear Friends,

With heartfelt appreciation, I extend this dedication to each of you, the foundational pillars of my life's journey. Your enduring support, unconditional love, and shared values have shaped my character and guided my path.

Your commitment to my education has been a guiding light, offering clarity in moments of uncertainty and strength in times of adversity. Through your encouragement and wisdom, I have learned the importance of resilience, integrity, and empathy.

You have been my steadfast companions on this voyage, enriching my experiences with laughter, companionship, and shared memories. From childhood adventures to the challenges of adulthood, your presence has been a source of comfort and joy.

Chapter 1

Introduction

TFL has eight means of transport that attend today roughly nine million people. It covers a region of 1,572 square kilometers, spanning over 1,500 kilometers of track, making it one of the largest and busiest urban transportation networks in the world. Over its rich history, the Transport for London (TFL) system has played a pivotal role in shaping the city's growth and connectivity, facilitating the daily lives of commuters and residents alike.

Although this study's significance lies not in the use of a public dataset or familiar graph methods, it might sound presumptuous to claim revelations about the Transport for London (TFL) system. The intricate approximations required to model it accurately are beyond our available resources. However, guided by a blend of common sense and creativity, this project tackles the formidable task of analyzing a highly efficient and complex transportation system.

At its core, our primary objective is straightforward: to assess the relevance of edges within the TFL network. But before we delve into the methodology, an essential question arises: relevance for whom?

While assessing the relevance of edges within the TFL network, we must consider whether this relevance pertains to the individual user or the community at large. It's indisputable that the agency responsible for a public transport system should prioritize the collective perspective. However, for the individual user, reaching their final destination is the central objective.

The chosen methodologies are instrumental in discerning between individual and collective perspectives within the transportation network. The intricate interplay between individual needs and collective outcomes is illuminated through our analytical approach.

The transportation network and its traffic are deeply intertwined, presenting a challenge in distinguishing between traffic generated by purposeful displacement and that driven by convenience. Factors such as distance, vehicle frequencies, speed, and traffic density represent only a fraction of the complexity inherent in this system.

The methodology employed for the distress ranking begins with a wrong assumption that all commuters seek the same criteria when choosing their route. While some prioritize environmental friendliness and may prefer biking, others may face constraints that limit their reliance on the least polluting means of transport. For financially constrained individuals, ticket prices play a significant role in decision-making, while comfort considerations, such as seating availability and Wi-Fi access, may also influence route choices.

By recognizing these diverse preferences and constraints, our study does not aim to capture the complexity of commuter decision-making processes within the transportation network. Given the need for an arbitrary metric to proceed with this study, we have chosen the least travel time as the basis for evaluating route selection by individual users. This metric aligns with the universal goal of minimizing time spent in the transportation system.

By defining the least travel time as our primary metric, we establish a common benchmark for assessing route preferences. This allows us to evaluate how traffic would respond to hypothetical disruptions and assess the severity of their impact on people's lives. Ultimately, understanding the implications of these disruptions is crucial for enhancing the resilience and efficiency of the transportation network.

An alternative approach will focus on exploring local correlations and the weight heterogeneity of edges within the network. Without disruption simulations, we will rely on statistical tests to infer the relevance of edges and filter out those that carry less information about the system. By adopting this approach, we aim to map the most popular edges in terms of normal functioning, thereby shedding light on the network's efficiency under optimal conditions. Unlike disruption-focused analyses, this approach allows us to prioritize the network's inherent efficiency understanding, without concern for alternative routes' reparatory efficiency.

While both aspects are crucial to the Transport System Operator, it's essential to acknowledge the inherent limitations of the results presented here. Whether constrained by approximations, methodology, or dataset limitations, our findings are inevitably restricted in scope. However, these limitations will be thoroughly addressed in the modeling chapter.

For a more faithful and representative model, it's imperative to consider much more complex questions. By doing so, we can enhance the accuracy and reliability of our analyses, providing deeper insights into the functioning of the transportation network.

Understanding what drives commuter decisions is pivotal in shaping the direction of system growth. This involves considering factors such as convenience, cost, environmental impact, and comfort. Additionally, optimizing the user experience and minimizing the system's costs, including CO2 emissions, travel time, and operational expenses, are key objectives.

Predicting passenger flows and their variance according to various factors, including hour, weekday, month, year's season (including school vacations), and sporadic events, is essential for effective planning and resource allocation.

To address these challenges, our study will consider three network infrastructures weighted according to the number of commuters and vehicle frequencies across six time periods: "Early", "AM Peak", "Mid Peak", "PM Peak", "Evening", and "Later". Additionally, we will factor in station distances and travel times for a comprehensive transportation system analysis.

1.1 The problem

The myriad of possible models for the TFL system prompts the first question: how do we represent this intricate network? Each station represents a node, and each rail connecting them is an edge. The simplest model would assign weights based on the distance between stations, but we can adjust these values according to the analytical objective of the model. Time granularity is crucial, especially when evaluating operational guidelines or daily traffic patterns. Variations in traffic conditions can influence vehicle speed, further emphasizing the relevance of time granularity in our analysis. The periods referenced in our analysis reflect the degree of approximation in our results.

Our analysis aims to comprehensively explore this dataset and uncover insights from operational and commuter perspectives. The latter considers mobility and travel quality, weighing values like time, cost, and comfort. Meanwhile, operational perspectives aim to allocate resources effectively for a comprehensive, isonomous, and efficient service.

One might envision a perfect metro system as one that interconnects all stations seamlessly. However, increasing the number of segments may lead to decreased journeys' popularity and increased congestion, despite the absence of topological or resource

limitations such as energy consumption. The congestion inevitably leads to time delays, impacting the system's efficiency.

The idea of a system that optimally meets user needs is constrained by our understanding of commuters, which is not the primary objective of this study. Instead, we simulate and evaluate different layers of the system to understand its mobility service in terms of efficiency, redundancy, and commuter behavior. We break down each mode of transport into three components: traffic, infrastructure, and operations.

This breakdown allows us to map segment congestion, envision infrastructure or operational improvements, and assess vehicle usage efficiency and route allocation. By adopting a suitable time granularity, we divide the network into three rudimentary subnetworks: infrastructure, operational, and traffic, each recalibrated according to specific indicators detailed in a separate section.

It's essential to acknowledge that there's no ideal network perfectly tailored to accommodate traffic. Firstly, while networks are built in response to traffic demand, they inadvertently encourage traffic increase. Secondly, the ideas of efficiency and redundancy are concurrent. A hypothetical fully connected network may be the most efficient by directly connecting stations. However, on the other hand, there is no matching redundant efficiency in case of a segment disruption, inevitably causing a decrease in service quality.

Chapter 2

Data collection

2.1 Data Sources

The study was possible due to the TFL Open Data initiative, which has promoted positive implications [1]. The entire data used in this study comes from the TFL. However, they come from two different sources: TFL Unified API and the Project NUMBAT.

The Unified API provides TFL's live information to registered developers. This project used the TFL API to collect infrastructural and operational information from the transport systems. Here, it is defined as infrastructural all the information coming from static network elements as stations or connections. Each Station is, for instance, composed of Latitude & Longitude and a NaptanId. Operational information refers to the systems management and its dynamic functioning. The vehicles travel respecting Lines, routes, and timetables. Moreover, the operational information varies according to the yearly seasons as scholar period, holidays, and the days of the week. After all, it does make sense to acknowledge the cycles of people's routines.

The Unified API allows registered users to access the transport system information. The "Application ID" and an "Application Key" are provided to be embedded in the HTTP requests addressed to the endpoints, organized to accommodate a wide variety of data. The retrieved data are presented in JSON format and had to be parsed before being stored in pandas data frames. The data collection step is founded on Client classes that effectively validate and easily access different endpoints to retrieve the desired information. The Client Class code was inspired by the GitHub project TfL-python-api. The specificity of this project required a comprehensive study of semi-structured response files (JSON) for multiple endpoints before creating the respective request functions for each of them (see section B.1).

The NUMBAT project stands as a comprehensive multi-rail demand dataset tailored for Transport for London (TFL). It furnishes detailed statistics about usage and travel patterns across TFL railway services. Leveraging ticketing data sourced from smartcards and gateline entry & exit tallies at each station, NUMBAT significantly augments the sample size compared to prior datasets. Moreover, it employs timetable-based assignments, thereby enhancing coverage across lower-frequency segments of the network and generating more nuanced outputs.

In conjunction with the modeling techniques explicated in the accompanying chapter, this dataset facilitates the evaluation of demand profiles, aids in service planning, and supports performance measurement. Specifically, the dataset encapsulates travel demands during typical autumn weekdays (Monday through Thursday), Fridays, Saturdays, and Sundays, each segmented within 15-minute intervals. Notably, it encompasses "Station Link Flows" and "Service Frequency" data, pivotal in determining edge weights, while "Origin & Destination by Mode of Transport" data informs estimates of passenger origin and destination probabilities along each edge, crucial for calculating disruption impacts.

Similarly to the TFL Unified API, the NUMBAT project is divided into Monday-To-Thursday (MTT), Friday, Saturday, and Sunday traffic profiles. Therefore, only MTT and Sunday data will be considered with the objective of preserving information consistency when creating a model that gathers different data sources.

2.2 Data Categories

The Transport For London (TFL) is one the most extensive and efficient current transport grid. The TFL counts on eight modes of transport and supports a population of almost nine million. Those modes of sum 697 lines, which may contain multiple routes. A route can be defined as a sequence of stop-point connections. Two stop-points can be connected by more than one mode of transport, line, or route.

To efficiently evaluate the mode of transport, it is important to bear in mind the difference between Lines & Edges. As further discussed in the Modelling chapter, a mode of transport can have different *Lines* connecting the same origin-destination stop-point pair, presenting different speeds, loads, and/or vehicle frequencies. To efficiently measure how a passenger can move from point "A" to point "B", naturally it is necessary to consider all Lines involved between "A" and "B". An *Edge* here is the concatenation of lines for a given pair of stations connected, where the resultant speed is the average weighted by the traffic of each involved line.

The speed magnitude is a key performance indicator (KPI), which composes other KPIs to evaluate desirable traits such as robustness and efficiency. Some indicators are time-variant and others are not. This fact motivated the sub-categorization of the collected data. Three data categories are deemed suitable for this particular study: infra-structure (instrument), Operational (service offer), and Traffic (service demand).

2.2.1 Infra-structure

The infra-structure data refers to stations and their connections. This information concerns the network design and is immutable, therefore requiring certain studies as it can be easily adjusted over days or weeks. The infra-structure set the displacement possibilities. Supposedly, stations and their connections are built in the face of the traffic's evolving needs. Furthermore, new stations are built into the rigid infrastructure to expand commutation possibilities and better accommodate the dynamic urban demand. Undeniably, the system expansion requires significant financial and time effort among other intangible obstacles, justifying a meticulous study for correct investment to better address the community interest. And graph theory concepts may come in handy.

This study aims to elucidate the significance of the segments beyond their mere volume, employing methodologies that account for station connectivity (degree) and strength (overall weight of their connections). Additionally, the concept of "Betweenness Centrality", which focuses solely on infrastructure, is employed to gauge the influence of a node in facilitating information flow.

The passenger flow is neither uniform nor fixed, it drastically changes along the day. In other words, urban dynamics condemn a connection significance in one moment and better appreciate it in the future. A valid reason why the operational management of the system must factor in scheduled maintenance and disruptions, which may necessitate flow redirection — an aspect that will be simulated and comprehensively discussed in subsequent chapters. Hence, the inflexibility of infrastructure should not be equated with simplicity when it comes to understanding the utility of segments vis-à-vis the overall performance of the network.

During the time of this study, Transport for London (TFL) comprised a total of 55,389 segments distributed across eight modes of transport, as delineated in the following table:

Bus	Cable	DLR	Overground	River-Bus	TFLRail	Tram	Tube	Total
54,082	2	92*	218*	114	60	71*	750*	55,389

Table 2.1: Number of connections by modes of transport.

Those segments connect over thousands of stations. All the stations in the United Kingdom are uniquely identified by a National Public Transport Access Node (NaPTAN). Two or more NaPTAN may share the same location in the case that one station grants access to two or more means of transport. On the other hand, a NaPTAN cannot have more than one *Latitude* & *Longitude*. Each NaPTAN represents a geographic location of access and exits for a given means of transport.

In summary, the TFL contains 32098 unique NaPTAN linked to a unique NaptanId. Find below the number of stations across the eight modes of transport.

Bus	Cable	DLR	Overground	River-Bus	TFLRail	Tram	Tube	Total
31,466	2	45	112	67	30	39	271	32,098

TABLE 2.2: Number of stations by modes of transport.

All the infra-structure information in this study is obtained through the TFL Unified API.

2.2.2 Operational Data

The operational information is here defined as what can be controlled by the system administration and, consequently, mutable in time. The routes and their timetables follow the traffic cycle and vary according to the day of the week and the time of the day. In this project, the operational data refers to train frequencies and speed. This is a piece of relevant information for the objective of understanding how many and how fast people are traveling. The Vehicle Load Speed is derived from the operation information coupled with the traffic, it will be defined in section 4.2. A high number of train departures (frequency) increases the chances of unexpected stops causing bottlenecks. On the other hand, low frequency can cause overload. The balance between train frequency and passenger can be translated into target train load, used to guide operational decisions for a more efficient transport grid.

The TFL Rest API provided the route's timetable, which enabled the travel time estimation for each segment. Following the timetable, a route's segment is traveled at different speeds depending on the period of the day. The timetable is composed of interval identities. Each interval identity reports the time at which the vehicle must perform at a given route's segment at a specific period of the day. For a matter of simplicity, due to the high processing cost and low relevance to ranking the segments, it was adopted a constant velocity for each vehicle, calculated from the segment distance and its most common interval identity.

The TFL presents over 600 lines across the eight means of transport. The table below display the number of of lines and routes for the means of transport concerned in this study:

	DLR	Overground	Tube	Total
Lines	1	6	11	19
Routes	10	24	64	104

TABLE 2.3: Number of access nodes by modes of transport.

A route is exclusively formed by an origin and destination. A line can be formed by one or more routes, inflicting one or more origin-destiny pairs.

In summary, the operational data encapsulates the functioning of the system and the decisions taken to optimize resources, thereby maximizing the utilization of infrastructure. This optimization is geared towards accommodating traffic and facilitating population mobility, with the overarching goal of making public transport systems more intelligent, environmentally sustainable, and safe.

2.2.3 Traffic

In accordance with the Freedom of Information Act (FOIA) and TFL's information access policy, Project NUMBAT has made available passenger journey data for the year 2019. The traffic data is sourced entirely from Project NUMBAT, utilizing ticketing data from smart cards and gateline entry/exit totals for each station. This dataset boasts a significantly larger sample size compared to other datasets and employs timetable-based assignments, enhancing coverage of lower-frequency parts of the network and generating more granular outputs. It aids in understanding transport system usage on different days of the week across various sub-networks, assuming an ideal train schedule.

Contrary to the TFL Unified API, Project NUMBAT provides information via a relational database rather than semi-structured data schemes, necessitating standardization efforts to correlate nomenclatures and abbreviations across different table sources. Although the NUMBAT dataset does not encompass all of TFL's means of transport, it contains affluent and essential information for modeling traffic flows on the London Underground, the London Overground, and the Docklands Light Railway.

The primary focus of this project is to model the efficiency of the public transport system and its temporal variations. Therefore, traffic is segregated into daily splits, acknowledging significant differences between weekdays, Saturdays, and Sundays. Furthermore, for each day, the traffic is aggregated into six periods: Morning, AM peak,

Midday, PM peak, Evening, and Late. The NUMBAT project offers a 15-minute granularity, allowing for a more detailed analysis of traffic dynamics. In this study, however, the enrichment rendered by a 15-minute granularity does not pay off the required computational effort. The period offers a good balance between the computational effort and the traffic dynamism. The period's composition is described in the table below:

Period	Range	Length
Morning	05h00-07h00	2 hours
AM peak	07h00-10h00	3 hours
Midday	10h00-16h00	6 hours
PM peak	16h00-19h00	3 hours
Evening	19h00-22h00	3 hours
Late	22h00-00h30	2,5 hours

TABLE 2.4: Timeband description for the NUMBAT Project.

The periods adopted cover 19.5 hours of the day, with indicator values resulting for each period hourly averaged to ensure that period length does not influence indicator magnitude. All time-variant information follows the period time frame, including Passengers by Origin & Destiny, Passengers by Segments, and Vehicles' Frequency, all crucial for efficiency and robustness models.

The Journey's Origin and Destination are essential information for the Robustness model. This data is used to create a statistical inference of the *Origin&Destinations* for the passengers of a given segment. The table 2.5 displays the number of traveled journeys for each means of transport:

O&D	DLR	Overground	Tube
Count	1,766	5,072	55,451

TABLE 2.5: Number of journeys from 2019 *O&D* NUMBAT dataset.

Naturally, the passenger count is proportional to the number of journeys, inherited from the network structure. The number of passengers per means of transport is displayed below:

Day	DLR	Overground	Tube
MTT	376,527	597,397	4,986,901
Fridays	376,161	599,595	4,986,901
Saturdays	256,380	452,757	3,639,598
Sundays	203,331	298,436	2,504,473

TABLE 2.6: Number of passengers from 2019 *O&D* NUMBAT dataset.

Finally, the chart below illustrates the distribution of passengers by period, emphasizing the utilization of Origin & Destination information in the robustness model, further elaborated in the methodologies section.

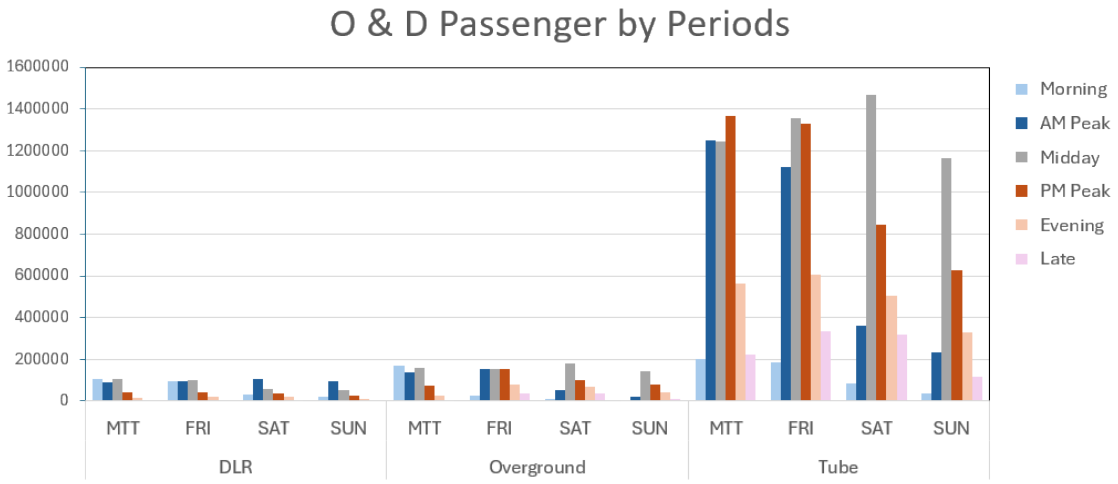

FIGURE 2.1: Plot of passenger numbers from 2019 *O&D* NUMBAT dataset.

Chapter 3

Methodology

3.1 Backbone Extraction

The disparity filter serves as the primary backbone extraction methodology adopted to examine the Transport For London (TFL) systems. The goal of backbone extractions is to reduce the original graph while retaining maximum information. A useful analogy would be the Principal Component Analysis algorithm, commonly employed in dimensional reduction scenarios. Consequently, backbone extraction techniques prove significantly beneficial in elucidating the structural characteristics of complex networks. With this objective in mind, these techniques have been applied to abstractions of London's intricate transportation system, represented as bi-directional weighted networks. Subsequently, comparing the results obtained from the techniques applied to transport systems, with different complexities will facilitate a more comprehensive understanding of the structure of the networks involved, as well as the efficacy of each applied technique.

The filtering methods for the backbone extraction are mainly divided into two categories: global and local.

3.2 Local Backbone Extraction

3.2.1 Disparity Filtering

The disparity filter is a local backbone extraction method [2] and it considers the local distribution of weight to edges. It aims to discern the extent of an edge's involvement at each given node, particularly in a model where the edges surrounding a single node are distributed heterogeneously.

For instance, let's consider a directed edge x_{ij} connecting the departing and destination nodes i and j. The edge's weight is defined by ω_{ij}, representing the network distance, the speed, traffic, etc. Each node has a strength value defined by the sum of the weights of the departing edges. The strength of the node i is represented by s_i. The edge x_{ij} involvement is defined by its weight ω_{ij} divided by the node strength s_i of its departing node.

$$I_{ij} = \omega_{ij}/s_i. \tag{3.1}$$

In a network with randomly distributed weights, the involvement fluctuation is characterized by the function Y_i. In the literature on complex networks [3], the function Y is recognized as a disparity measure and standard used in various fields and expressed by:

$$Y_i = \sum_{j \in \Psi_i} I_{ij}^2, \tag{3.2}$$

where Ψ_i is a set of neighbors of node i.

The involvement heterogeneity, for a given node i with k neighbors, is characterized by the function $\gamma_i(k)$, as demonstrated below:

$$\gamma_i(k_i) = k_i Y_i(k_i) = k_i \sum_{j \in \Psi_i} I_{ij}^2 \tag{3.3}$$

Assuming all edge weights are equal, each edge involvement I is $1/k_i$. Hence, it follows a homogeneous distribution with $\gamma_i(k_i) = 1$. Conversely, when one edge weight is responsible for the total node strength, we have $\gamma_i(k_i) = k_i$, representing maximum heterogeneity. In real systems, it is observed that $\gamma_i(k) \propto k^a$ and the exponent is close to 1/2.

The disparity filter method employs a null model to assess the edges' relevance. The null model admits a probability density function with anomalous weight fluctuation. Let's assume "$k-1$" points dividing the interval [0,1] into "k" equal sub-intervals. The sub-interval lengths represent the expected involvement value. The probability density function[2] is denoted as

$$\rho(x,k) = (k-1)(1-x_{ij})^{k-2}dx. \tag{3.4}$$

The statistical relevance of an edge is represented by α_{ij} and its values lie in the interval [0,1]. The complementary values correspond to the involvement of the neighbors of the edge x_{ij} (SI_{ij}). Thus, the statistical relevance of an edge is expressed below:

$$\begin{aligned} \alpha_{ij} &= 1 - SI_{ij} \\ &= 1 - \int_0^{I_{ij}} \rho(x, k_i) dx \\ &= 1 - (k_i - 1) \int_0^{I_{ij}} (1 - x)^{k_i - 2} dx, \end{aligned} \tag{3.5}$$

where the term "$(k_i - 1)$" represents the number of neighbors of the edge x_{ij}, the term "$(1 - x)$", inside the integral, represents the involvement of x_{ij} edge's neighbors, and the exponent "$k_i - 2$" represents sub-intervals occupied by the x_{ij} edge's neighbors.

The statistical relevance of an edge arises from the heterogeneity of its involvement in relation to other local involvements (adjacent edges). It's worth noting that each node in the network with a certain degree k is compared with the null model corresponding to their degree.

Below, a directed graph with randomly weighted edges is utilized as a toy example. On the left-hand side, edges are color-coded based on their weights, while on the right-hand side, they are colored according to their statistical relevance (α). In Figure 3.1, one can discern the global approach, represented by the weights ranking, from the local approach, represented by the statistical relevance ranking.

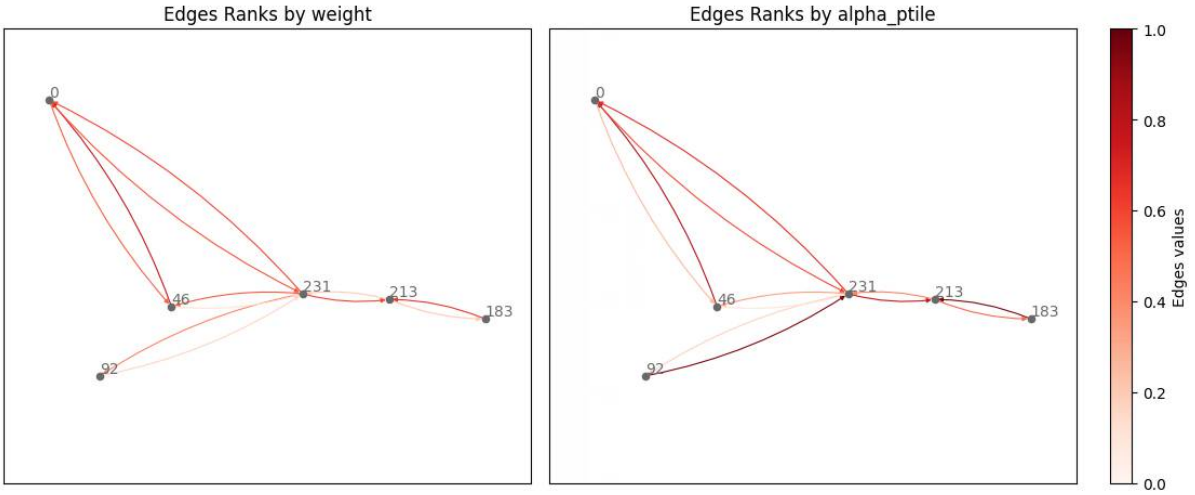

FIGURE 3.1: Edges rank comparison plot - toy example.

For instance, the link $(92, 231)$ exhibits a reverse priority between the global weight and the local weight fluctuations. This can be explained by considering that the direction

$92 \rightarrow 231$ represents the total strength of node 92. However, since node 231 has multiple departures, the statistical relevance of $231 \rightarrow 92$ is weakened in the local approach. All edge values are described in the table below.

Percentile Rank	**Edge**	**Weight Rank**
(1) 0.958	(183 - 213)	(2) 0.639
(2) 0.958	(92 - 231)	(11) 0.159
(3) 0.833	(46 - 0)	(1) 0.762
(4) 0.750	(231 - 213)	(3) 0.602
(5) 0.667	(231 - 0)	(4) 0.598
(6) 0.583	(0 - 231)	(5) 0.557
(7) 0.500	(213 - 183)	(9) 0.197
(8) 0.417	(213 - 231)	(10) 0.192
(9) 0.333	(231 - 46)	(6) 0.540
(10) 0.250	(0 - 46)	(7) 0.530
(11) 0.167	(231 - 92)	(8) 0.412
(12) 0.083	(46 - 231)	(12) 0.099

TABLE 3.1: Edges rank comparison table - toy example.

The Kendall rank correlation, which measures the strength and direction of association between two ranked variables, is 0.52. This suggests a moderately strong positive relationship between the global weight and local filtering rankings. Although the techniques are noticeably different, this result was expected since both methodologies consider the weight magnitude.

The disparity filter method highlights structures neglected by the global threshold filter. It illuminates the nuanced interplay between edge weights and network topology, offering invaluable insights into the complex dependencies inherent in real-world networks. The disparity filter is a versatile multi-scale reduction algorithm that can be used as a preliminary indicator of the presence of local heterogeneities.

Service equity is a critical prerequisite for public transport systems. It must adapt to the unique characteristics of each region. The Journal of Transport Geography published studies presenting strategic network indicator and node-place-design models to classify tube stations in London [4]. The study based on identifying station groups with relatively high network criticality and relatively low node-place-design score is of potential value. Although not applied in this study, the disparity filtering method would come in handy due to its efficacy in identifying local fluctuations.

While the disparity filter method offers significant advantages in extracting network backbones by exploiting local heterogeneity and correlations among weights, it does come with certain limitations. One notable drawback is its reliance on systems characterized by strong disorder where the null model does not represent normalcy. The null model considers a homogeneous weight distribution, which would highlight fluctuations that are common in a disordered system. This restricts its applicability to networks with specific structural characteristics.

Additionally, the methodology's effectiveness may diminish when applied to networks lacking pronounced heterogeneity in edge weights. Furthermore, while the disparity filter excels in reducing the number of edges while preserving weight and node integrity, its impact on network topology may vary, potentially altering properties such as degree distribution and clustering coefficient. As such, careful consideration must be given to the suitability of the disparity filter for a given problem domain, and it may be prudent to explore combinations of different techniques to achieve optimal results.

3.3 Global Backbone Extraction

3.3.1 Thresholding Filtering

The thresholding method for backbone extraction in complex networks involves establishing a threshold value to filter out edges, retaining only those that exceed the specified criterion, thereby ranking the edges based on their weight. While this method is relatively straightforward to implement, it operates independently of the network's structural context, focusing solely on edge weights or specified criteria to identify the backbone structure. Consequently, although thresholding effectively prioritizes edges according to their attributes, it may overlook important structural properties of the network, such as clustering patterns or hierarchical organization. Therefore, it is important to recognize that the resulting edge ranking may not fully capture the nuanced interplay between network structure and edge importance. Hence, integrating additional network analysis techniques that consider the relational context may offer a more comprehensive understanding of edge ranking within complex networks. The Robustness indicator, which assigns weights to edges, can provide the relational context crucial for thorough network analysis, offering insights into the network's resilience to perturbations.

3.4 Indicators

3.4.1 Efficiency Indicator

An important goal of network management operations is to enhance the overall efficiency of a transport system, considering the dynamic nature of traffic patterns and the constraints imposed by available resources. To assess this efficiency, this study proposes a customized metric to evaluate the connections representing the transportation between stations. This proposed indicator integrates three fundamental aspects of the transport system: operational guidelines, traffic dynamics, and infrastructure intricacies. The image below provides a visual representation of the resulting efficiency indicator.

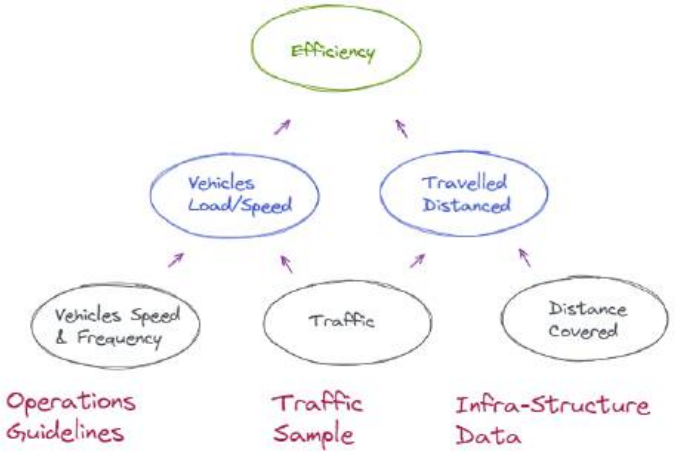

FIGURE 3.2: Key performance indicators layers.

Though purely theoretical and without aspirations beyond serving this backbone extraction study, the efficiency indicator proposed by this study integrates elements of speed, distance, and load to assess the edges. The efficiency of each edge can be calculated using the following expression:

$$\begin{aligned} Efficiency &= (Train\ Speed) \times (Train\ Load) \times (Total\ Travelled\ Distance) \\ &= (Train\ Speed) \times \left(\frac{Passengers}{Hourly\ Train\ Frequency} \right) \times (Passengers \times Distance) \\ &= \frac{Train\ Speed \times Passengers^2 \times Distance}{Hourly\ Train\ Frequency}. \end{aligned} \tag{3.6}$$

Despite the varying lengths of day periods, efficiencies are calculated based on the total extent of each period and averaged hourly. This approach facilitates comparisons of edge efficiency across different times of the day. It's crucial to acknowledge that only

passenger traffic and operational frequencies of trains fluctuate throughout the day. For simplicity, the speed of an edge remains constant throughout both the day and the week.

3.4.2 Robustness Indicator

In contrast to the localized approach exemplified by the disparity filter, the robustness model adopts a global methodology to evaluate the transport system network. The objective here is not to determine the superior method, but rather to comprehend the unique perspectives offered by each approach. However, it is crucial to select an analytical method that not only respects the network structure but also facilitates edge ranking, thereby enabling a comparison with the localized approach. The robustness indicator, when coupled with the Thresholding methodology, fulfills these requirements by maximizing the utility of available data while remaining applicable within the domain of public transportation systems.

The proposed method aims to assess station connection robustness, which essentially evaluates how effectively the network can navigate the absence of a given edge by providing commuters with alternative routes. In theory, any alternative path resulting from an interruption should be less efficient than the original path. However, certain exceptions are detailed in the section Robustness Model 4.7. It is also important to note that the study defines the optimal path, for simplicity, as the one with the shortest journey time. Using the shortest path algorithm, we determine the increase in travel time if a particular edge is removed. By mapping all origins and destinations affected by the disrupted edge, we can calculate the total increase in travel time for each origin-destination pair.

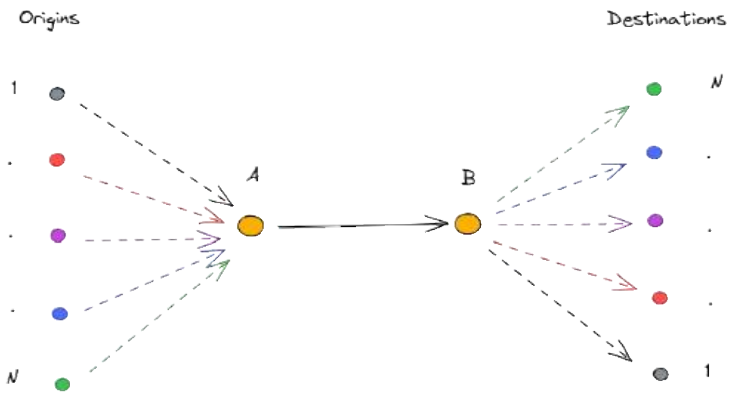

FIGURE 3.3: Origin-destination pairs passing by stations A and B.

However, total increased time fails to adequately represent service impact. Leveraging the aggregated traffic data for each origin-destination pair (sourced from NUMBAT OD Traffic 2.2.3), the likelihood of commuters utilizing a specific origin-destination pair along a given edge is calculated. Consequently, the total increased time due to edge disruption is weighted by the likelihood of each origin-destination pair, an approach proposed by this study, which will be referred to henceforth as the Estimated Disruption Increased Time (EDIT).

$$\begin{aligned} EDIT &= \frac{\Delta t_1 \times (Passengers\ OD_1) + ... + \Delta t_N \times (Passengers\ OD_N)}{(Passengers\ OD_1 + ... + Passengers\ OD_N)} \\ &= \frac{\sum_{k=1}^{n} \Delta t_k \times (Passengers\ OD_k)}{\sum_{k=1}^{n} (Passengers\ OD_k)} \end{aligned} \tag{3.7}$$

Where:

- $Passengers\ OD_N$: the number of passenger that perform the journey n,
- Δt_N: the time increased to perform the origin-destiny n after disruption.

The EDIT facilitates the computation of the Estimated Lifetime Impact (ELTI), which this study proposes to signify an edge's impact for any given period if its traffic is known. This metric is stratified by day periods and weekdays to refine the robustness model of a dynamic network. The granularity of ELTI follows the periods' aggregation, ensuring that the indicator accurately reflects variations in traffic patterns throughout different times of the day and week.

$$ELTI = EDIT \times (Passengers\ of\ the\ Edge) \tag{3.8}$$

It's worth noting that some stations have a degree of connectivity equal to one, implying no alternative routes exist to reach the original destination apart from the edge under analysis. To address this limitation, a workaround is to assume that passengers can traverse the edge's distance on foot in case of disruption, with time calculated as the edge distance divided by the walking speed. Another pertinent factor omitted is the time passengers spend when changing lines. The robustness model necessitates several simplifications to remain viable, including the arbitrary choice of time for evaluating the optimum path – many passengers may prioritize comfort or cost, for instance. The traffic data essential for this model comes from two independent sources, which are divided to mitigate concerns regarding data anonymity and privacy policies. To incorporate both

sources, one dataset is used to calculate the origin-destination pair probabilities for each edge, and then the other dataset is utilized to determine the lifetime impact.

One advantage of this model lies in its consideration of individual passenger journeys, providing a single-user perspective that complements the efficiency indicator utilized in the local method, which is more operationally oriented as it considers train load and distance. Although a tabular global method such as thresholding is employed to rank edges, the overall result incorporates network structure since the shortest path algorithm is utilized in ELTI calculation. Bearing this in mind, a similarity in rankings between the global thresholding method and the local disparity filter method is noticed when the ELTI indicator is employed to weigh the edges. The scenario and the results are detailed in the section 7.3.

3.5 Rank Comparison

The edge's importance rank serves as the output of each tested backbone extraction methodology outlined earlier. To quantitatively compare the ranking across various periods of the day and days of the week, a methodology capable of assessing convergence and divergence is essential. The aim is to gauge how edge priorities fluctuate throughout the day, considering the dynamic nature of traffic within a public transport network.

The Kendall Comparison methodology proves efficient for pairwise rank comparisons, making it instrumental in this study for assessing the correlation between different periods of the day, days of the week, and the backbone extraction methodologies mentioned above. The aim is to comprehend how the resulting models capture distinct elements of a complex network, thereby enabling more accurate analysis of complex systems. The comparisons and their outcomes are elaborated throughout Chapter **??**.

3.5.1 Kendall Correlation

The Kendall correlation value reveals the degree of agreement or disagreement between the rankings. A Kendall tau coefficient close to 1 indicates a strong positive correlation, signifying high agreement in edge importance rankings across different periods and days. Conversely, a coefficient near -1 suggests a strong negative correlation, indicating substantial disagreement in rankings. A coefficient around 0 implies no significant correlation, implying that the rankings are independent of each other.

After implementing the Disparity Filter and Robustness methods and computing the ranks for comparisons, Kendall's tau coefficient [5] is calculated to measure the

concordance between the two rankings. This involves tallying the number of concordant and discordant pairs of ranks across the ranks and applying the following formula:

$$\tau = \frac{\text{Number of concordant pairs} - \text{Number of discordant pairs}}{\text{Total number of pairs}} \tag{3.9}$$

Where:

- Number of concordant pairs: the count of pairs of data points that have the same order in both datasets,
- Number of discordant pairs: the count of pairs of data points that have different orders in the two datasets,
- Total number of pairs: calculated as $n(n-1)/2$, where n is the data points in each dataset.

While the Kendall correlation offers valuable insights into the consistency of edge importance rankings, it has certain disadvantages. Kendall's tau does not account for the magnitude of differences between ranks, focusing solely on their order. This may overlook situations where the magnitude of differences between ranks is essential for interpretation. Additionally, it is sensitive to tied ranks, which can skew correlation values, particularly in small sample sizes.

On the other hand, the Kendall correlation is robust to outliers and does not assume any particular distribution of data, making it suitable for non-parametric analysis of ranked data. Furthermore, Kendall's tau is intuitive to interpret, measuring the proportion of concordant pairs relative to discordant pairs. Its simplicity and versatility make it a widely used tool for assessing the similarity or dissimilarity of rankings in various fields, including transportation network analysis.

Chapter 4

Modelling London Transport's System

4.1 The Transport For London -TFL

The TFL System comprises a comprehensive grid of more than 20 thousand stations and 50 thousand connections and operates according to periods of the day, days of the week, and scholarly seasons. This fact significantly increases the challenge of building a faithful model. The approach initially splits the part of the original system into three sub-networks corresponding to one mode of transport. Unfortunately, the available information is not consistent across all modes of transport, making it unfeasible to conduct the same analysis for each of them. As mentioned, Project NUMBAT provides data only for railway services such as the tube, overground, and DLR. Therefore, the study will concentrate solely on these modes, which allow for the creation of a weighted directed network.

Each mode of transport will be treated independently, with any inter-mode connections considered to constitute a single passenger journey. Consequently, during disruption simulations, passengers will either need to walk the distance covered by the disrupted edge or seek an alternative path within the same mode of transport. While this approach diverges from real-world scenarios where subsystems complement each other, it allows for a focused analysis within each mode. However, it's worth noting that an ideal systemic ranking, transcending individual modes of transport and enabling overall segment comparisons across the entire system, would offer a more comprehensive perspective. This aspect will be discussed further in Chapter **??**. Additionally, it's important to acknowledge the approximations made regarding travel time and distance between stations.

The "TFL Unified API" meticulously maps the travel time of vehicles across multiple intervals. Despite the detailed nature of this information, this project considers the time represented by the "Interval Id" equal to zero, as it is the most commonly used interval identity throughout the day. Additionally, the project does not account for the time spent during connections, including both the time spent in transit between platforms and the waiting time until the next train arrival.

Regarding distance calculations, the project employs Euclidean distance measurements, which may not fully account for potential trajectory curves or slopes along the route.

In terms of traffic analysis, the project determines the "Origin & Destination" for every passenger based on the shortest travel time. This approach establishes a relationship between edges and their respective "Origins & Destinations," which is essential for the robustness study, where disruptions are simulated.

During disruption simulations, alternative paths that pass through new stations are calculated, while the starting and ending stations of the journey remain fixed. However, this approach involves a slight approximation, as passengers' final destinations may not necessarily be stations but rather addresses that could be reached by other stations. In other words, disruptions may affect both the starting and ending stations of a given journey.

4.2 TFL Transports Systems

Three transport systems are modeled in this study: the Tube, Overground, and DLR. While these models are developed under the same premises, they exhibit varying complexities due to differences in infrastructure size, traffic volume, and operational lines. In these models, nodes represent stations, and edges represent connections between adjacent stations within the Transport for London (TFL) transport systems. It's important to note that the connections used in the models accurately reflect the real connections between stations. For instance, let's consider the segment between "Acton Town" and "Earl's Court" stations in the Tube system.

FIGURE 4.1: Tube Map London segments.

Figure 4.1 illustrates the segment, indicating that both the Piccadilly line (blue) and the District line (green) connect the aforementioned stations. It's worth noting that these lines' trains run over different railways, except for some segments where they merge, for instance, between "Hammersmith" and "Barons Court" stations.

The base-graph model gathers in a unique model all the necessary information and it is used as a base for the subsequent models where the studied methodologies are applied. It is interpreted as the line representation of the mean of transport. While trains performing the Picadilly Line and District Line possibly share the same railway, the base-graph represents their lines.

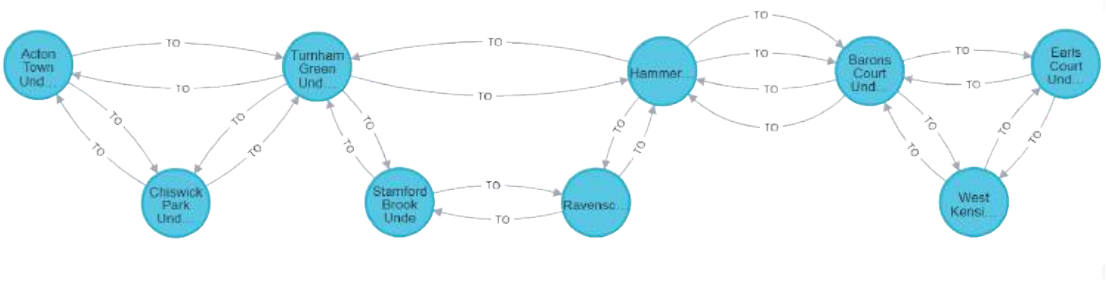

FIGURE 4.2: Base-graph model of the Tube Map London.

In the tube system, a route refers to the origin and destination of a train, encompassing the entire journey between two points. A line can accommodate multiple routes, and these routes can be adjusted by the system management as needed. When evaluating infrastructure management, it's sufficient to represent only one edge between two stations to capture the connectivity within the network.

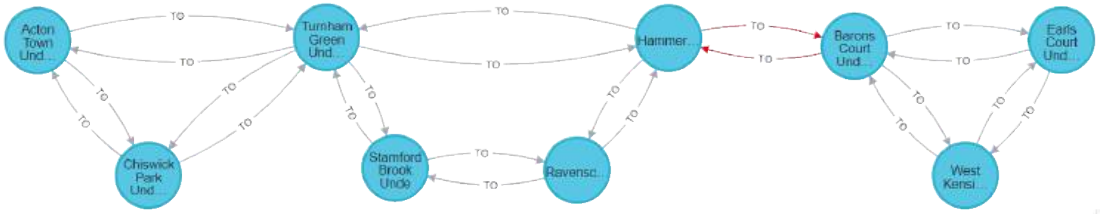

FIGURE 4.3: Equivalent model of the Tube Map London.

The equivalent connection traffic and train frequency are calculated by aggregating the values from its component lines. Similarly, the speed is determined by weighting the

speed of each component line by its respective load. The average load of a line is defined as its traffic divided by the number of trains that have departed for a specified period.

$$Average\ Train\ Load = \frac{Number\ of\ Passengers}{(Train\ Capacity) \times (Number\ of\ Train\ Departured)}. \tag{4.1}$$

Since the train carriages of TFL adhere to a standardized design for each mode of transport, with consistent volume and seat numbers, the model relies on relative load metrics.

$$Average\ Train\ Load \propto \frac{Number\ of\ Passengers}{Number\ of\ Train\ Departured}. \tag{4.2}$$

This equivalent model serves as the foundation for generating a series of models that incorporate different indicators as edge weights across various periods of days and days of the week. Each model is constructed using a directed graph that reflects the direction of journeys, with node degrees calculated based solely on departing edges.

4.3 Transport Model Composition

The objective of applying the filtering methodologies and indicators is to examine how the size and complexity of a network can impact backbone extraction. To illustrate the spectrum of complexities across the systems studied in this research, consider the following network counts for each modeled network:

1. **Tube**: 268 stations, 710 line segments, and 614 directed edges;
2. **Overground**: 111 stations, 215 line segments, and 215 directed edges;
3. **DLR**: 45 stations, 92 line segments, and 92 directed edges.

The network models are categorized into time-variant and time-invariant indicators. Time-variant indicators, such as traffic, necessitate the creation of multiple networks per mode of transport, considering different days of the week: Monday to Thursday daily average (MTT), Fridays (FRI), Saturdays (SAT), and Sundays (SUN). Within each day, six periods are considered: Morning (05:00-07:00), AM Peak (07:00-10:00), Midday (10:00-16:00), PM Peak (16:00-19:00), Evening (19:00-22:00), and Late (22:00-00:30). Conversely, for time-invariant indicators, a single network model is constructed, and

both local and global methods are applied for comparison purposes. Closing this gap necessitates the utilization of a graph database.

4.4 Graph Database Systems

Graph databases employ a specialized query language known as Cypher, which enables efficient data retrieval and manipulation. One of the primary advantages of graph databases is their suitability for visualization, allowing users to intuitively comprehend complex relationships through graphical representations. The base graphs are constructed using Neo4j, with all the necessary information for the subsequent graphs and further computation of methods stored in the elements' attributes.

4.4.1 Nodes

The nodes represent the stations and are composed of the following attributes: number of entries/exits per period, latitude, longitude, station name, and captained. Each station at the TFL system contains a unique ID named Naptanid.

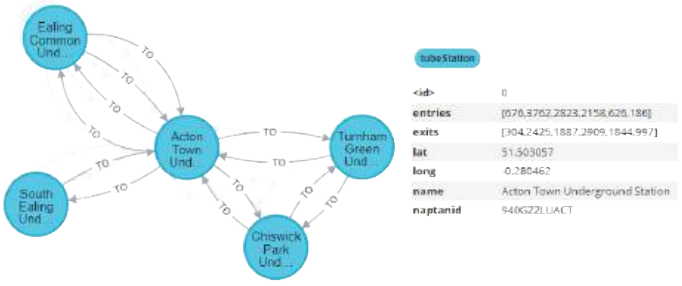

FIGURE 4.4: Station representation as nodes.

4.4.2 Edge

The edges represent the stations' connections and are composed of the following attributes: mode of transport, distance, speed (km/h), traveled time (in minutes), traffic, and number of trains departing by period.

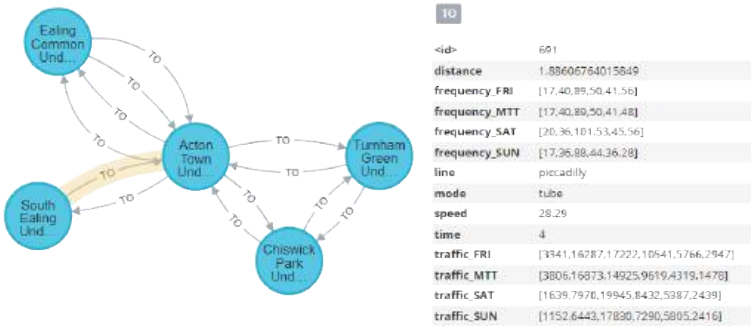

FIGURE 4.5: Connections representation as edges.

Neo4j utilizes Cypher as its query language and provides a robust platform for handling and exploring interconnected data. Its inherent graph structure allows for the smooth execution of graph algorithms, such as the shortest path algorithm utilized in the Robustness indicator. In essence, Neo4j serves as a versatile and efficient database platform, enabling the extraction of valuable insights from complex graph data.

4.5 Programming Frameworks

The local disparity algorithm could not be run in the Neo4j graph database. To do so, the methodology was implemented using Python programming language. The network models were exported from the Neo4J graph in a ".json" format and imported using the NetworkX library. The NetworkX stands as a fundamental framework for the analysis and visualization of complex networks, offering a versatile and comprehensive set of tools.

NetworkX facilitated the import and manipulation of the intricate network structures represented by directed graphs (digraphs). The library offers essential functionalities for this study, including algorithms for centrality measures, and the implementation of custom network algorithms. Moreover, NetworkX supports visualization and data manipulation capabilities, enabling users to generate clear and insightful representations of network structures. It achieves this through integration with popular libraries such as Matplotlib, Pandas, and SciPy.

For this study, NetworkX serves as a cornerstone allowing the application of the disparity filtering methodology, and facilitating the development of a theoretical model, as well as simulations of real-world phenomena. Its open-source nature fosters collaboration and innovation within the scientific community, while its extensive documentation and active user base ensure accessibility and support for researchers at all levels of expertise.

4.6 Disparity Filter Algorithm

The Disparity Filtering algorithm is an adaption of Paco Nathan's implementation of the publication Extracting the multiscale backbone of complex weighted networks [2]. The original code is designed with modularity and flexibility in mind, allowing the customization and extension of the functionality as needed. As such, the algorithm was adapted to run through each graph model exported from the Neo4J in the ".json" format and plot graph cuts for the backbone analysis.

The script begins by loading KPIs graphs from a Neo4J database into a NetworkX framework. The KPIs graph are distance, speed, traffic, distance traveled, loads, efficiency, and robustness. The last five indicators are applied to six time periods and four days. The total number of disparity ranks is 122 graphs per mean of transport. By analyzing different KPIs, exploring network dynamics across various spatial and temporal scales, and adapting visualization techniques to specific research questions or objectives.

All indicators are normalized using the Min-Max method. The Disparity filter algorithm requires that all the edge values are in the range 0 to 1. The Min-Max normalization method was chosen to sprucely distribute the values across the range, reducing the variance. It preserves the edges' relative magnitudes, which is crucial to the application of the disparity filtering algorithm.

Once the graphs are imported, the disparity measures (α) are calculated for every edge based on specified key performance indicators (KPIs). The significance value becomes an edge attribute. After all the edges' significance is calculated, the percentile score is also computed and stored. The percentiles allow the extraction of multiple backbone scales of the original network. The graph cuts are performed at various percentiles of significance and plotted using the Matplotlib library to gain the first insights into network dynamics.

The Disparity Filter Algorithm has a modular design that leverages its flexibility towards different indicators and network structures considered here. The algorithm has proven to identify edges with disproportionate influence on network dynamics. The visualizations offer intuitive representations of network dynamics, patterns, and critical routes, aiding the understanding and analysis of the transportation system, and facilitating targeted interventions to enhance system efficiency and resilience.

4.7 Robustness Model

The robustness model seeks to identify the edge's importance in terms of distress caused to the system users. The distress in this case is the additional waiting time if the edge does not exist. The passenger Origin & Destiny's information is crucial to come up with such an indicator.

Customer OD matrices are fundamental information required in public transit planning and operations analysis. The existence of systems with only entry controls, for example for buses, creates a need for methods of inferring the trip OD. Most transit OD matrix estimation methods are based on aggregate passenger counts. In contrast, there are methods based on individual passenger travel information as the destination inference algorithm [**?**], that takes advantage of the pattern of a person's consecutive transit trip segments and uses the next trip segment boarding location information to infer the destination of the prior trip segment. The NUMBAT Project, however, provides an OD matrix sparing us the effort of such computation.

The origin and destination traffic information is divided by days and aggregated by periods. However, it's important to note that this information may not equally encompass the entire system. For the DLR and Overground systems, the origin and destinations for the Late journey periods are not significant. Consequently, this will later impact the Late columns in the Heatmap for these means of transport.

Nevertheless, the robustness indicator requires a series of computations (as described in the methodology section). To do so, this study uses the Python language and, more specifically, the Neo4J library that allows access to the graph database.

For each origin destiny in the OD matrix in the NUMBAT data, it is computed the shortest path between a source and a target node in the model using the Dijkstra Source-Target algorithm. Time is set as the cost variable (relationshipWeightProperty) for finding the paths. The Dijkstra Shortest Path algorithm is x²available through the Neo4j Graph Data Science (GDS) library, which provides extensive analytical capabilities centered around graph algorithms. Below is an example of the Cypher command:

```
MATCH (source:overgroundStation), (target:overgroundStation) WHERE source.
    naptanid='910GBATRSPK' AND target.naptanid='910GWCHAPEL' CALL gds.
    shortestPath.dijkstra.stream('overground',{sourceNode: source, targetNode:
    target, relationshipWeightProperty: 'time'}) YIELD index, sourceNode,
    targetNode, totalCost, nodeIds, costs, path RETURN index, sourceNode,
    targetNode, totalCost, nodeIds, costs, path
```

Listing 4.1: Dijkstra Shortest Path Cypher query example

From the shortest path algorithm's results is formed a dictionary where the segments are the keys and their corresponding OD are inner dictionaries containing the traveled original time. After running the shortest path for all the ODs, we have the complete dictionary where all the used segments are stored as keys and all their corresponding OD is stored.

```
segments: {OD_1: [time], OD_2: [time], ··· , OD_n: [time]}
```

Listing 4.2: Segments OD Time Dictionary Example

After that, the rupture simulation for each dictionary key and the shortest path algorithm were re-computed for each of its OD sub-dictionaries, adding the delta increase time due to the disruption.

```
segments: {OD_1: [time delay], OD_2: [time delay], ··· , OD_n: [time delay]}
```

Listing 4.3: Segments OD Delays Dictionary Example

Naturally, nodes with a degree equal to one present a unique challenge as they can only be reached via their single edge. In such cases, the speed of the edge is replaced by 5 km/h, representing the walking speed of the user. The time delay is then calculated as the additional time required if the user were to walk the edge distance at this speed compared to the actual speed of the train. Interestingly, it was observed that not all shortest paths necessarily involve the fewest number of stations, as discussed in Appendix A, Section A.1. Traveling through more stations may increase the journey time, particularly if passengers need to change lines, as it results in more stop points for the train. While this issue could be mitigated by implementing a penalty cost, it was not included in this project, as the objective is not to precisely model the TFL system for practical use.

The estimated EDIT for the segment is computed using the formula **??**. Initially, the EDIT is derived from hourly averaged traffic data for each day and period, which is then utilized to generate the ELTI indicator. To ensure uniformity, min-max normalization is applied before integrating the robustness values into the graph edges and ranking them accordingly. It's important to note that the entire process incurs a computational cost that grows at least cubically, given that it involves quadratic processing for calculating the shortest path of the Origin & Destination matrix and subsequently recalculating them following each segment rupture.

Chapter 5

Network Analysis

This chapter provides a comprehensive analysis of network properties across various scales, revealing intriguing similarities and patterns. Despite differences in size, stations consistently exhibit a degree of two, indicating an optimal structure for managing passenger flow. Notably, the three networks display dual cores: an internal core with dense interconnections and a peripheral core with fewer links. The higher node betweenness in smaller networks highlights the central nodes' importance in determining efficient pathways and connectivity, given the limited path options available. Understanding these network properties provides insights into optimizing infrastructure design and management, ensuring smoother passenger flow and enhanced operational efficiency.

5.1 K-core Algorithm

The K-core algorithm is a method from graph theory used to identify cohesive subgraphs within a graph structure. It iteratively removes nodes with degrees lower than a specified threshold, denoted as 'k', until no nodes remain.

If 'k' is not explicitly chosen, the algorithm incrementally increases it until finding the maximum value that allows for a non-empty k-core subgraph. This process unveils the core structure of the graph, where nodes are tightly interconnected. The resulting k-core subgraph represents the most interconnected and stable portion of the original graph, often aiding in various network analysis tasks such as identifying central nodes or understanding community structures.

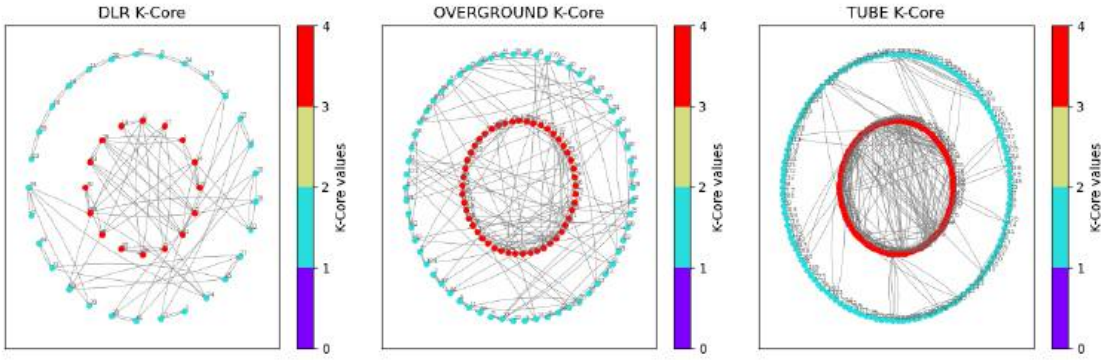

FIGURE 5.1: K-Core plots from base-graph.

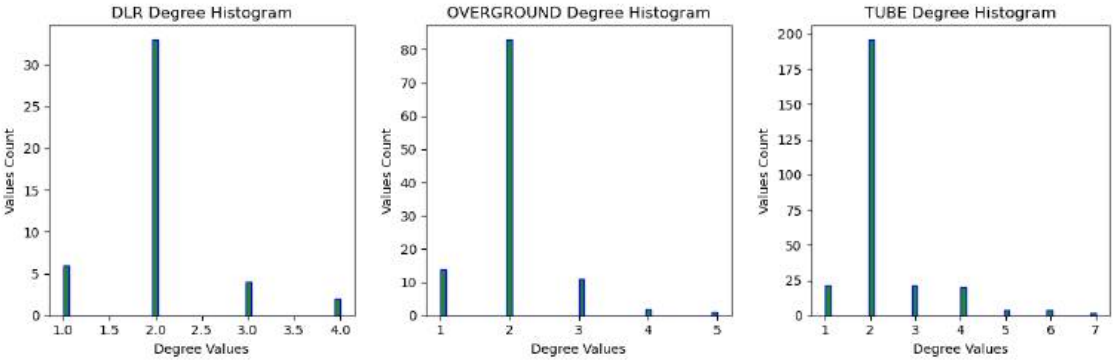

FIGURE 5.2: Degrees histogram

The findings highlight how nodes with higher core values tend to be more interconnected than those with lower values. In all three graphs analyzed, nodes consistently had the same core scores, primarily 2 and 4, due to similar degrees among most nodes. However, it's interesting to note that central and peripheral nodes connect differently across the network. When comparing the DLR and Tube systems to the overground system, there's a noticeable difference in connectivity between external and internal core nodes, with the former systems showing less connectivity in this aspect.

5.2 Between Centrality

Betweenness centrality measures a node's importance in facilitating communication across a network by quantifying its presence on the shortest paths between other nodes. The algorithms iterate over origin-destination pairs of nodes and attribute credit to intermediary nodes. Nodes with high Betweenness Centrality serve as crucial bridges, fostering connectivity and information flow.

This metric finds applications in diverse fields such as social network analysis and infrastructure management, aiding in the identification of influential nodes and understanding of network dynamics. Here we calculate the Betweenness Centrality based on the edges' distances.

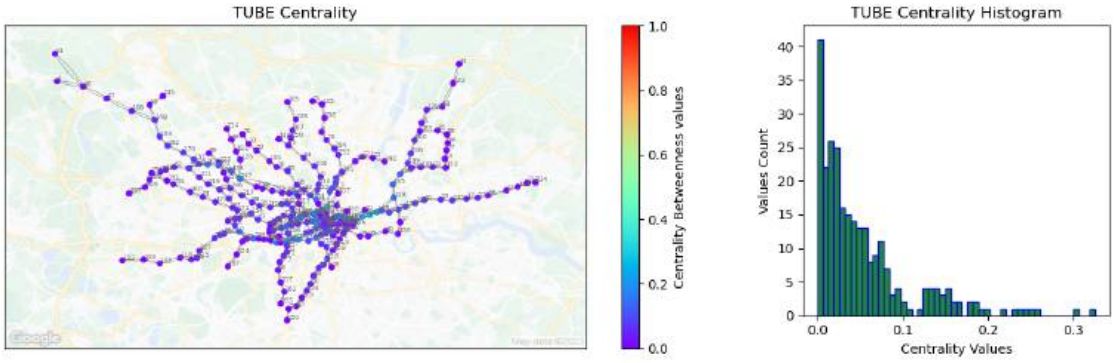

FIGURE 5.3: Tube betweenness centralities.

In the analysis of Betweenness centrality values, the DLR exhibits a greater dispersion of centrality scores compared to the Tube system, where most nodes cluster around values close to one. higher centrality values.

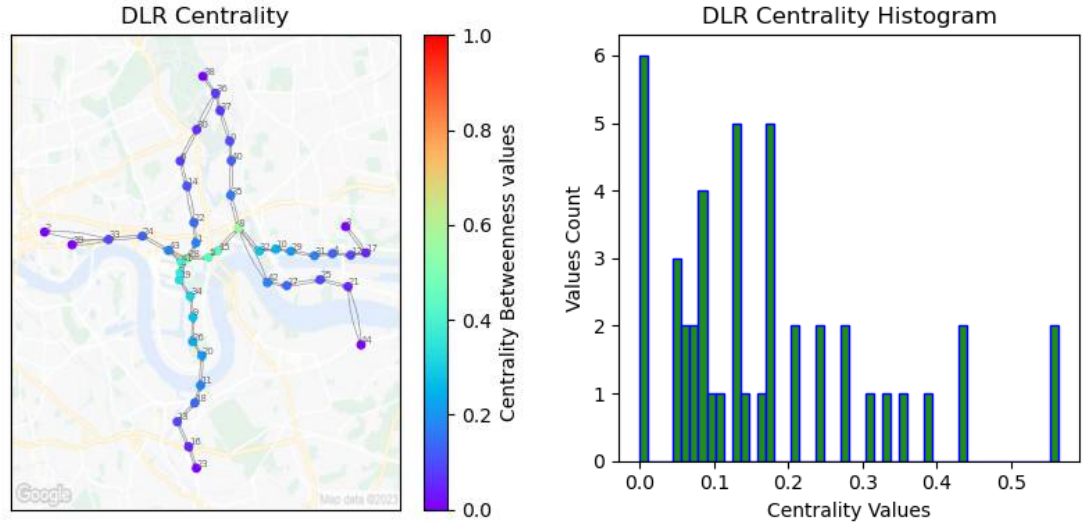

FIGURE 5.4: DLR betweenness centralities.

This disparity can be attributed to differences in network sizes. The extensive node count in the tube system diminishes individual node centrality, as fewer nodes are likely to participate in multiple shortest paths. Conversely, in the DLR, central nodes assume a pivotal role in system-wide connectivity, resulting in

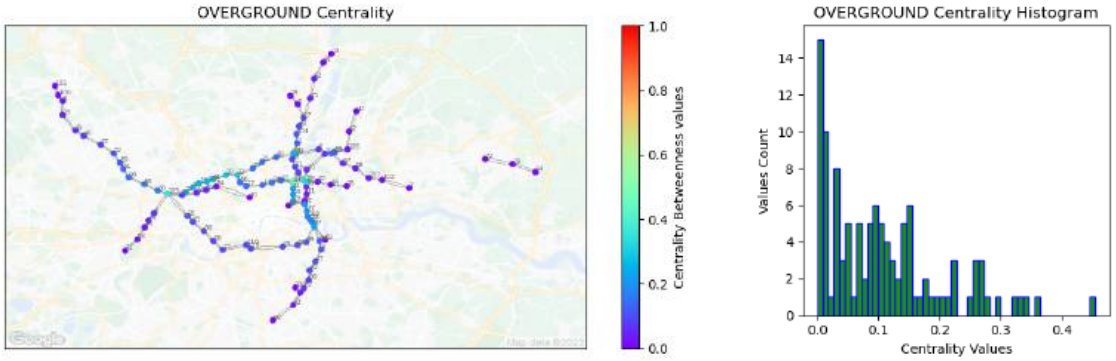

FIGURE 5.5: Overground betweenness centralities.

The overground system demonstrates an intermediate centrality value, consistent with its intermediate structural size.

Chapter 6

Transport System Analysis

This section examines data derived from transportation systems. Initially, it explores infrastructure and operational metrics like distance and speed. For all metrics, across networks of different sizes, the histogram reveals a more consistent distribution for the statistical relevance values (Alpha) than the more sparsely distributed normalized weight.

Subsequently, the focus shifts to traffic measurements. From the station's perspective, the entry and exit traffic plots display passengers' loading and offloading patterns, revealing nuances in flow directions when correlated with station geolocations. From the edge's perspective, the disparity filtering reveals flow directions indicated by the station entry and exit traffic. The histogram and heatmaps confirm the traffic inversion for certain periods. The thresholding methodology outranks edges with higher traffic, which are centrally located but fails to report the traffic inversion between antagonist periods faithfully.

6.1 Distance

The distribution of the distance across each network can be seen also in the histograms below:

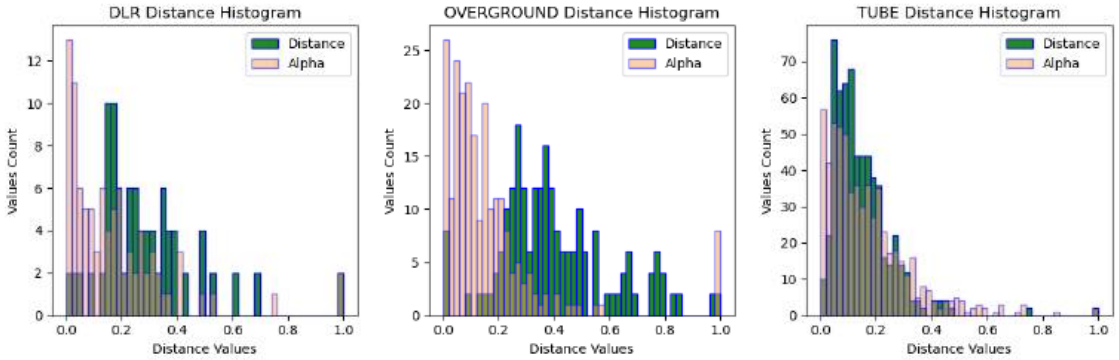

FIGURE 6.1: Distances histograms.

The edge's weight distribution is depicted in green, while the disparity relevance is represented in red. It is observable that station distances often diminish in larger networks, owing to increased station density. The disparity methodology assigns significance based on local heterogeneity, resulting in the majority of nodes exhibiting similar relevance values, typically close to zero. This methodology appears resilient to network size variations, as it evaluates edges consistently irrespective of node size.

The distances, however, between stations are independent of each other. This inherent attribute yields distinctive outcomes in backbone extraction compared to other edge indicators. Absent prior knowledge of the network, it is rational to prioritize edges covering larger distances as more significant. Disparity filtering emerges as an intriguing methodology for harnessing network structure, as evidenced by its impact on preserving the 50th percentile of the original network structure.

Upon employing the local filtering technique, a notable continuity in the lines becomes apparent. This observation may suggest a deliberate effort to establish connections between suburban areas and the central region, contrasting with the more scattered representation offered by the global thresholding technique. The Kendall coefficient for ranking compassion between the Disparity and Thresholding method for each mean of transport is shown below:

Ranking Comparison	DLR	Overground	Tube
Kendall Coef.	0,423	0,442	0,385

TABLE 6.1: Kendall coefficient for disparity and thresholding rankings.

Notably, this example, based on the distance indicator devoid of flow influence akin to traffic, underscores the application of local disparity in assessing an edge's importance. This assessment extends beyond mere connection distance to consider the local network

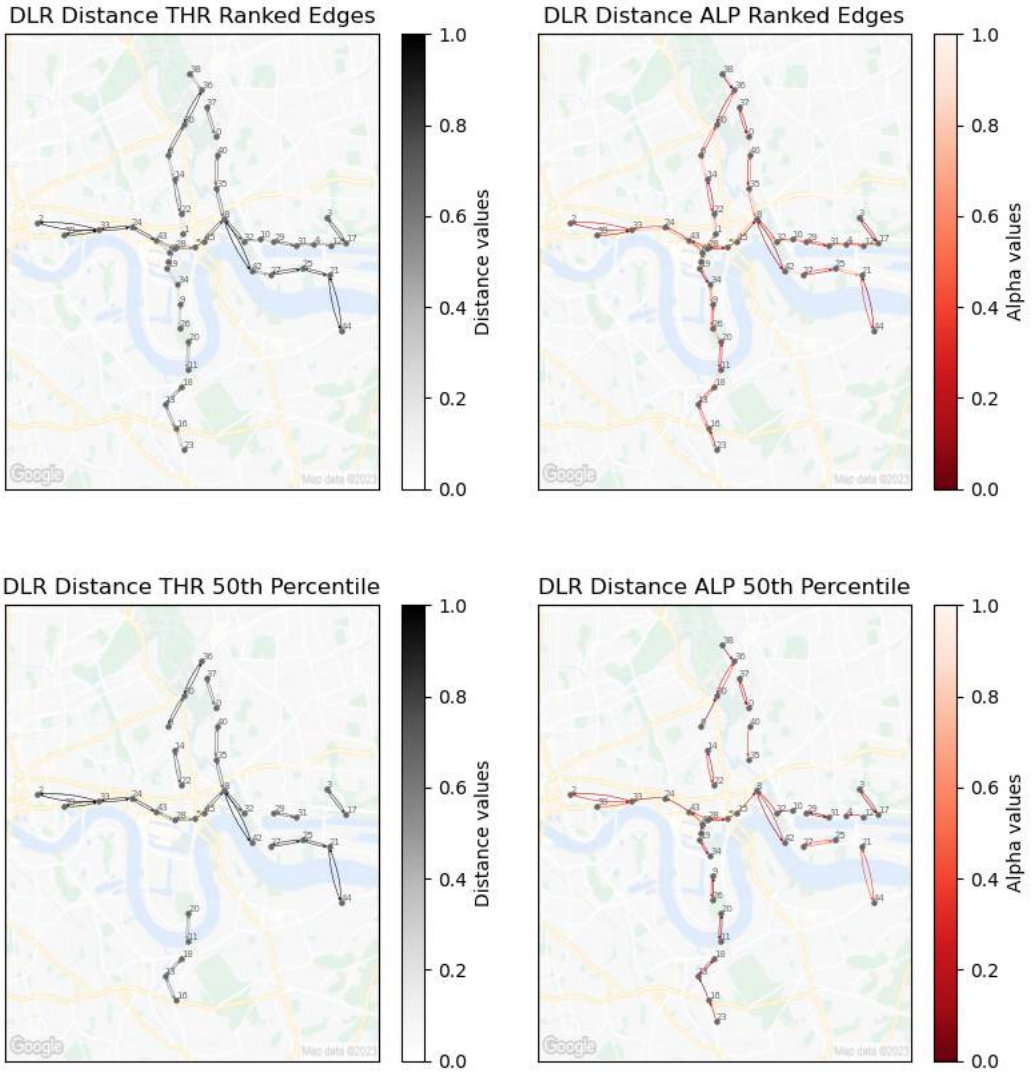

FIGURE 6.2: Backbone extraction comparison for edges' distance.

structure. Similar analyses were conducted for the Overground and Tube systems, with corresponding plots available in Appendix A, Section A.2.

6.2 Speed

The distribution of the speed across each network can be seen also in the histograms below:

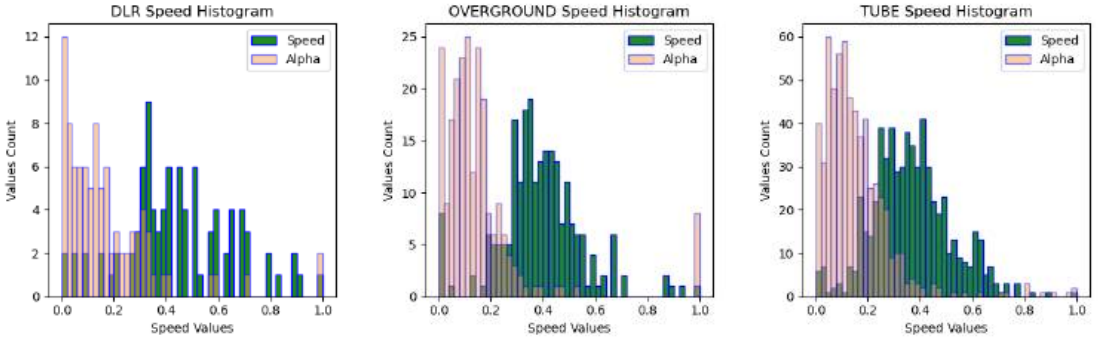

FIGURE 6.3: Speed histograms.

The speed distribution appears visually more even across all three systems compared to distance. This outcome is anticipated, given that speed is an operational indicator, unlike distance, which is contingent upon urban and topological variables. In terms of relevance, the distribution of edges within the systems is similar concerning speed and distance. However, how these relevances are distributed within the edges differs, as indicated by the ranking.

Following a similar approach to the distance exercises, backbone extractions are conducted using thresholding and disparity filtering methods. However, in contrast, both extractions now closely mirror the original structure.

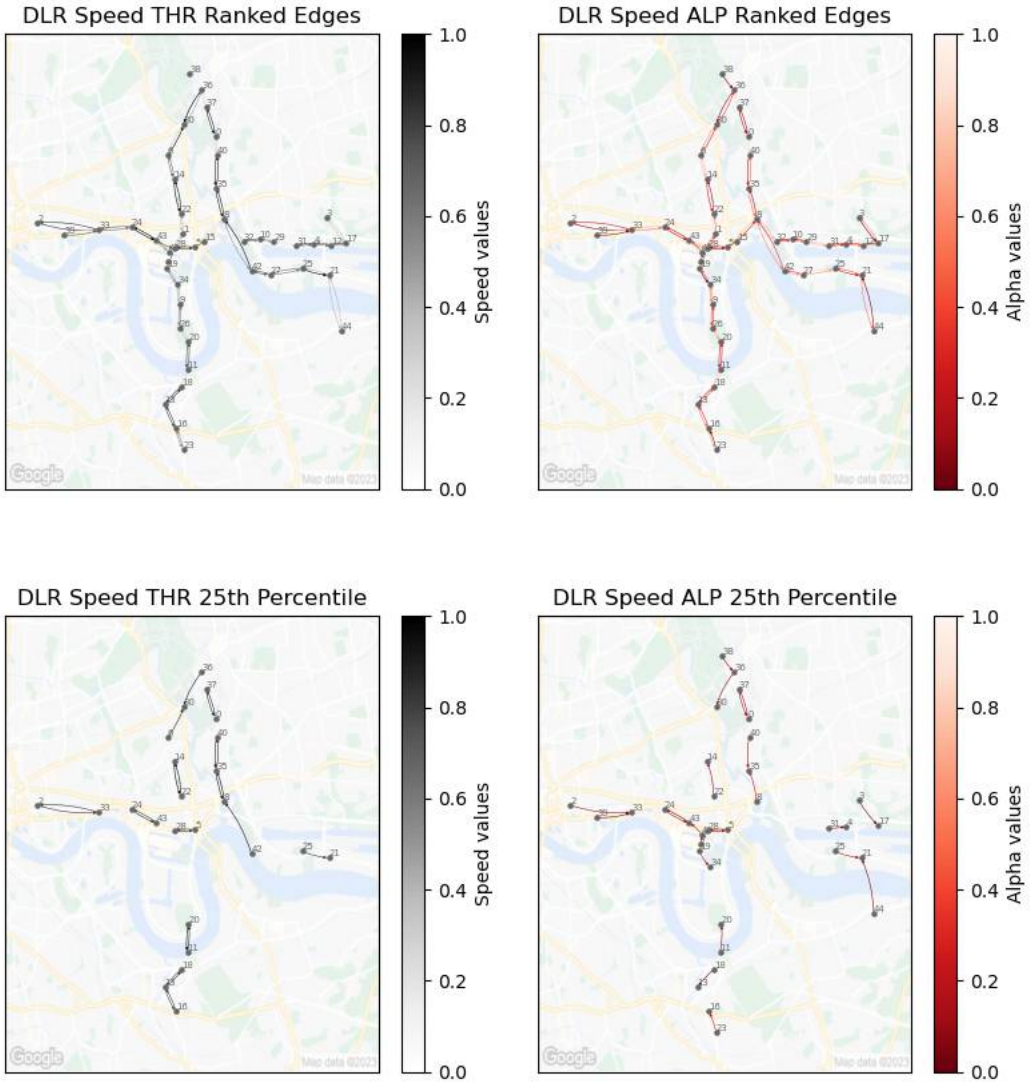

FIGURE 6.4: Backbone extraction comparison for edges' speed.

This phenomenon arises due to the average speed nearing the upper limit, resulting in a majority of edges meeting the thresholding condition. It is crucial to note that edges with lower values of α are deemed more relevant, whereas the opposite holds true when ranking by edge weights. Similar analyses were conducted for the Overground and Tube systems, with corresponding plots available in Appendix A, Section A.6.

Ranking Comparison	DLR	Overground	Tube
Kendall Coef.	0,468	0,387	0,450

TABLE 6.2: Kendall coefficient for disparity and thresholding rankings.

It is evident that both the Disparity and Thresholding methods, when applied to the speed indicator, yield higher Kendall coefficients compared to those derived from

the distance indicator for the Tube and DLR systems. Conversely, this trend is reversed for the Underground system. This discrepancy underscores that an indicator alone does not guarantee method convergence. It indicates that speed plays a different role in the Overground system compared to its role in the Tube and DLR systems concerning distance. In other words, there exists a divergence in values when considering speed versus distance allocation.

6.3 Traffic

Traffic can be interpreted more as an organic indicator, representing a factor that cannot be entirely planned. It offers insights into how users utilize the infrastructure in conjunction with operational guidelines to reach their destinations. Traffic information is typically divided into two components: Nodes and Edges.

6.3.1 Stations

The aggregation of passenger entries and exits at stations over various periods is stored as attributes within the nodes. The ensuing flow of passengers, whether entering (blue) or exiting (red) the station, is visualized across three distinct periods of the day. The size of each station's node reflects its relative magnitude compared to others. Additionally, histograms represent the distributions of entries and exits.

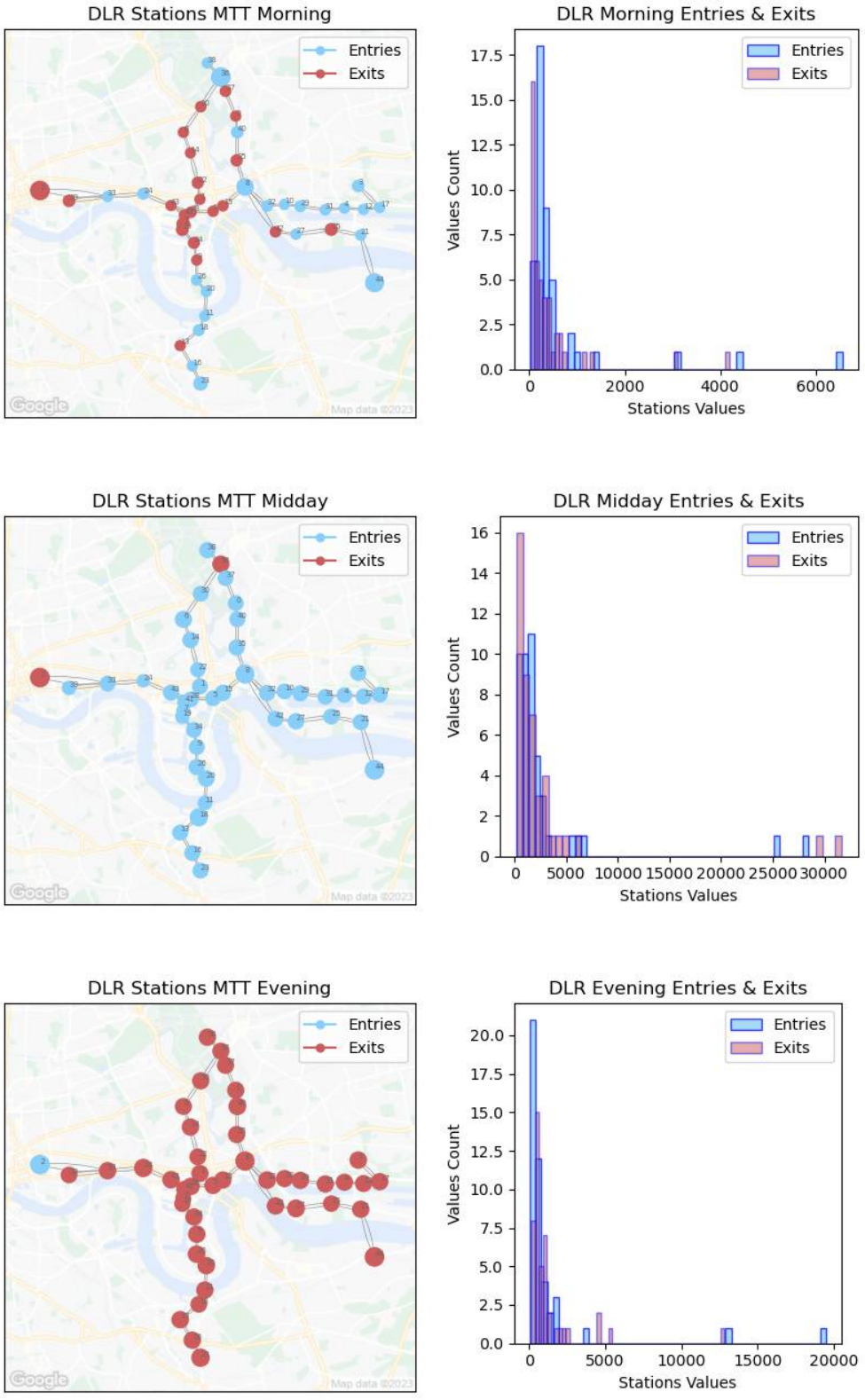

FIGURE 6.5: DLR station entries & exist plots and histograms for morning, midday, and evening periods.

The passenger entries and exits plots and histogram for the Overground and Tube systems are shown in Appendix A, Sections A.5.1 and A.5.2.

It can be inferred that transport systems experience an influx of traffic during daytime hours while experiencing a decrease during the night. Additionally, the direction of flow typically moves from suburban areas towards the city center in the morning, and conversely, from the city center to suburban areas during the night. This contrast between early and late periods is further supported by the histograms.

Interestingly, there are no significant differences observed in traffic patterns among the transportation systems. This outcome was anticipated, as all systems serve the same urban area.

6.3.2 Connections

Traffic inherently mirrors specific properties of the network influx. With passengers being transported between stations, it's logical to assume that neighboring stations would showcase similar traffic patterns that would require a local analysis instead of one that is based on global absolute values.

To facilitate methodology comparison, rankings for each of the six periods are computed, followed by the calculation of the Kendall Correlation Indicator. The heatmaps below illustrate the comparison between the methodologies:

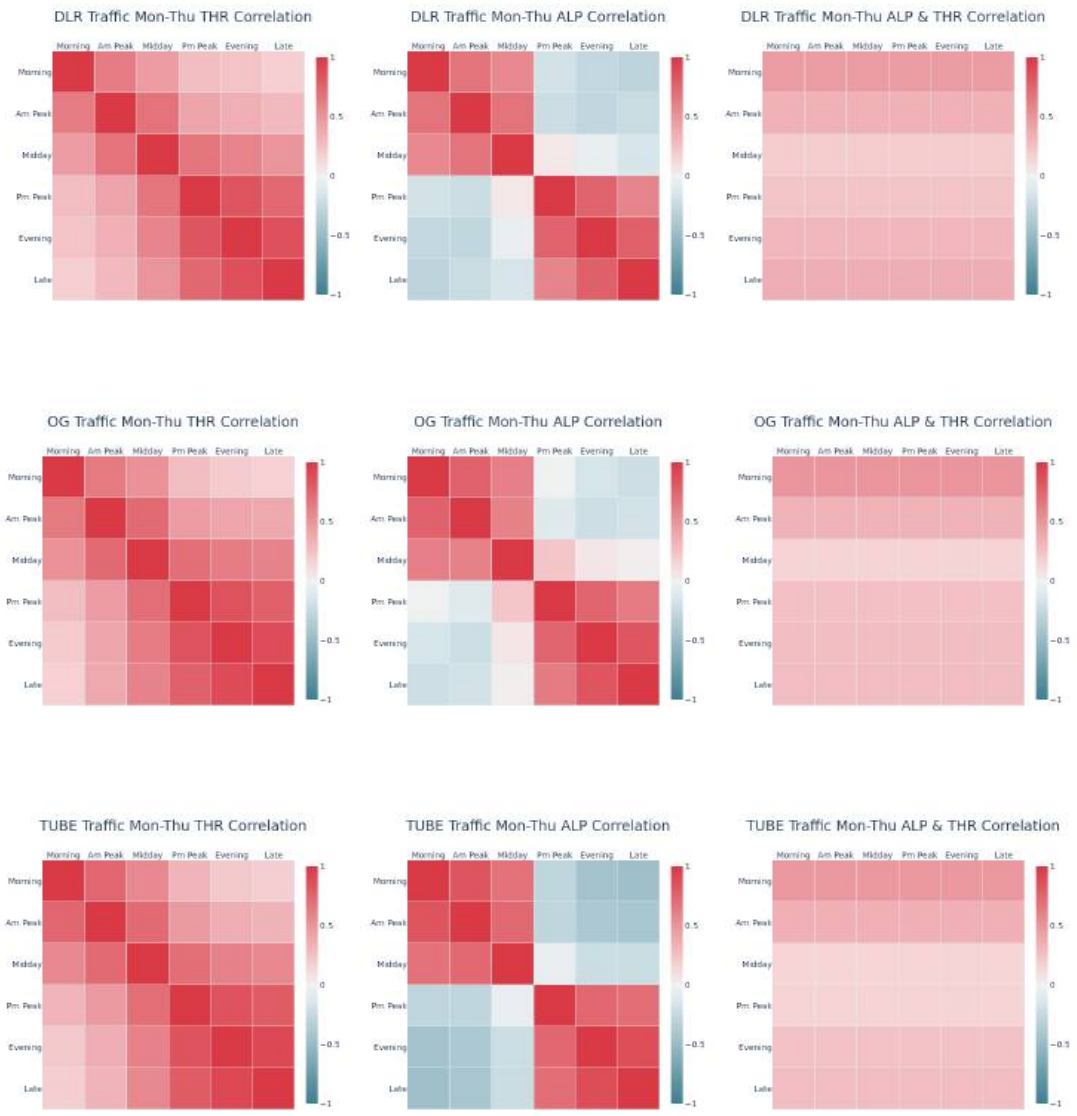

FIGURE 6.6: Traffic ranks heatmap across the methods.

While thresholding indicates independence between early and late periods, the disparity filter unveils negative correlations, suggesting that the ranking produced by the disparity captures inbound and outbound traffic with a patient transition throughout the day. As the traffic magnitude remains significant for disparity analysis, it is possible to remark a convergence between both methodologies. The third column of Heatmaps illustrates how the correlation between the two methodologies fluctuates across periods and days.

In terms of the backbone structures, the images below plot the edges belonging to the 50th percentile backbone for both methodologies for different periods of the day.

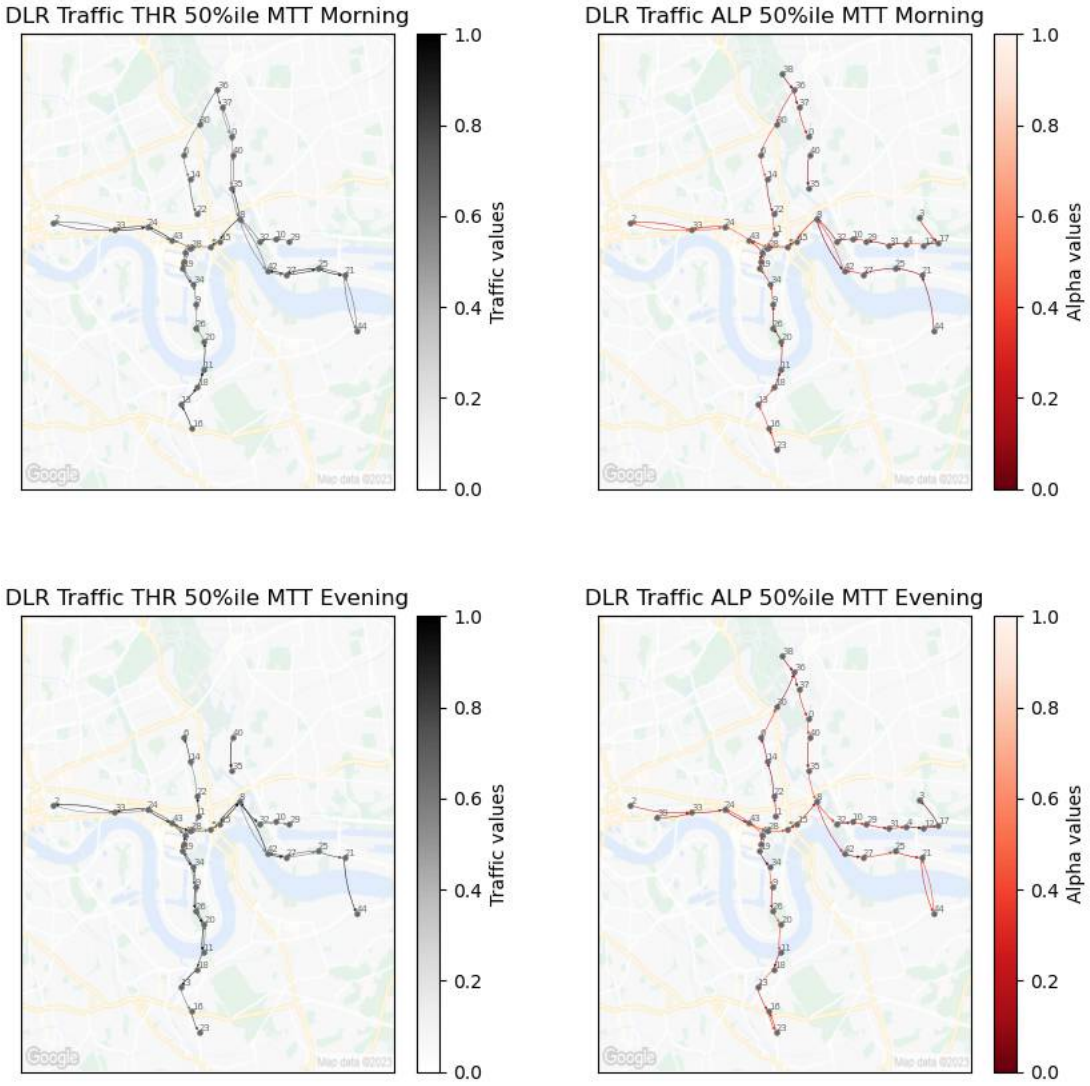

FIGURE 6.7: Traffic backbone extractions of DLR.

The Disparity filtering method showcased a more consistent backbone structure, notably with edges' directions reverting. Conversely, the Thresholding methodology failed to display the north side of its structure during the Evening period. The traffic backbone extraction plots for the Overground and Tube systems are shown in Appendix A, Sections A.4.1 and A.4.2.

While the traffic indicator encapsulates network properties within a transport system, it may not fully capture passenger flow, thereby reflecting edge relevance primarily in terms of traffic. The heatmaps provided below compare global and local extraction methodologies. Notably, the disparity filter proves adept at capturing traffic flow nuances and highlighting opposition between period flows.

The disparity filter considers the network's local structure when ranking edges, proving particularly effective for indicators devoid of network traces such as distance. As the indicator incorporates more network structure properties, such as traffic, its robustness increases, as explained later.

Utilizing the disparity filtering methodology for further network analysis, the traffic flow comparison between Monday to Thursday and Sunday for the DLR system is displayed.

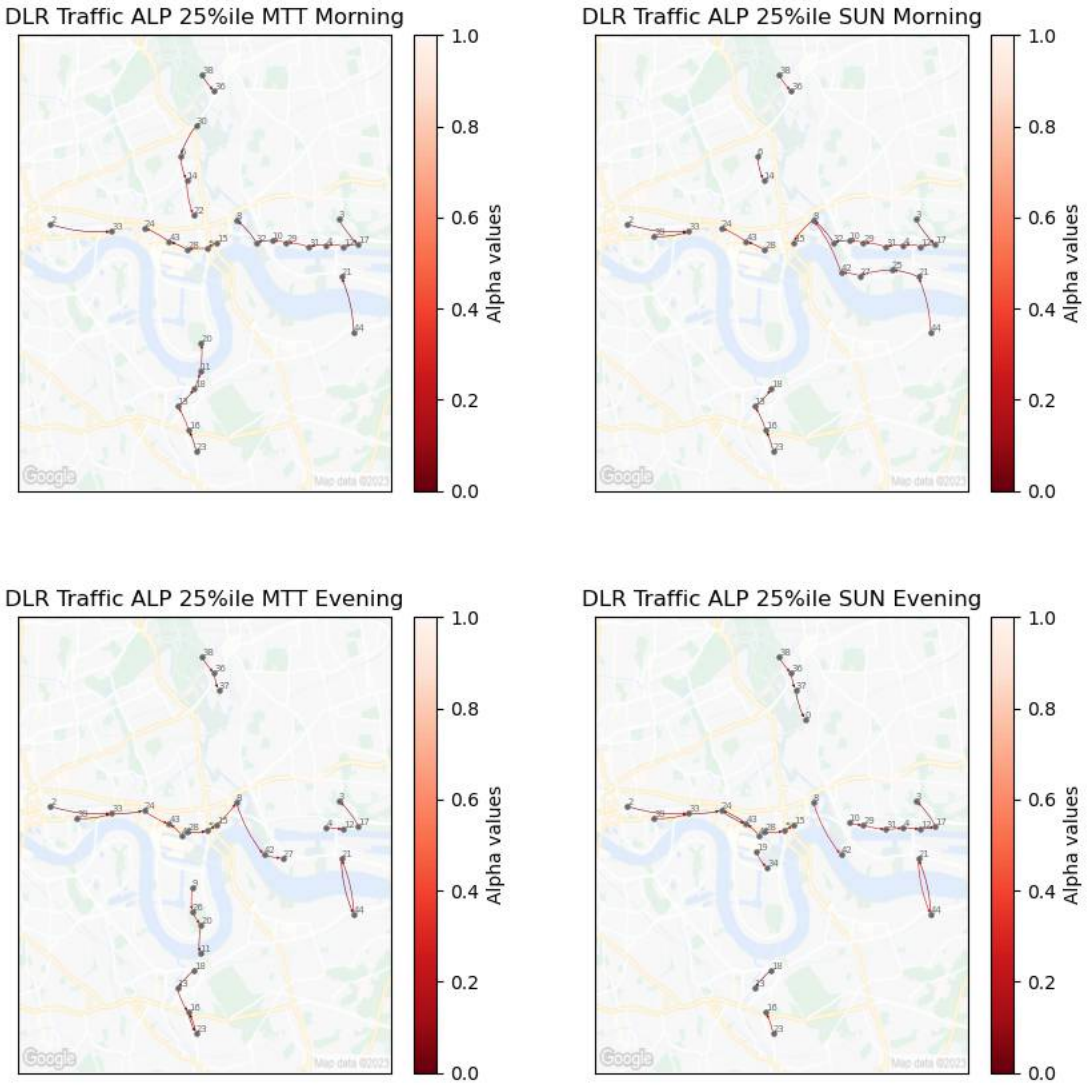

FIGURE 6.8: DLR MTT and SUN traffic backbone comparison.

No significant structural changes can be noticed among the days or periods. However, on both days, the traffic flow is reversed between the periods. The same be encountered for the Overground and Tube systems concerning all days are shown in Appendix

A, Sections A.6.1 and A.6.2.

The traffic reversion can be effectively highlighted through correlation heatmaps. Comparing different weekdays, the resulting heatmaps for the DLR system is shown below.

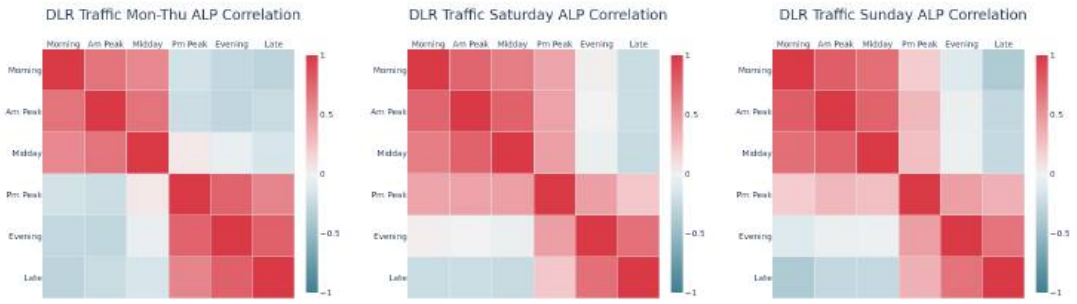

FIGURE 6.9: DLR traffic ranks heatmap across the periods.

Noticeably, the weekend days exhibit stronger flow similarities compared to weekdays. This observation can be further supported by analyzing the ranking across days. Although edge priorities exhibit a strong overall correlation across all days, the distinction between weekdays and weekends is clear. The heatmaps for comparing traffic periods of the DLR, Overground, and Tube systems across all days are depicted in Appendix A, Sections A.7.

FIGURE 6.10: DLR traffic ranks heatmap across the days.

The Sunday morning ranking order is less correlated to other mornings. However, the correlation between Sunday and other days increases for later periods in the day. The Heatmap figures for comparing traffic days of DLR, Overground, and Tube systems concerning all periods are shown in Appendix A, Sections A.8.

Chapter 7

Efficiency and Robustness Analysis

The efficiency and robustness indicators conveyed by this study aim to address the different concerns of the network, thus a low correlation is expected. It is observed that the correlation between the two indicators tends to decrease as the network size increases. Naturally, it is expected mainly because the ranking ordering possibilities are bigger for networks with more edges. However, the main reason is explained by analyzing the indicators separately.

7.1 Efficiency

Having compared the elements of infrastructure (distance), operation (speed), and traffic for each network, we will now explore how the aggregated indicator behaves throughout the periods and days for each network independently. The disparity methodology will solely play the role of ranking the edges. The objective is to understand how efficiency varies within each network, reflecting the challenges and opportunities in managing resources to provide a consistent and optimal service globally. The heatmaps below display how the efficiency ranks vary across the day for the three transport systems.

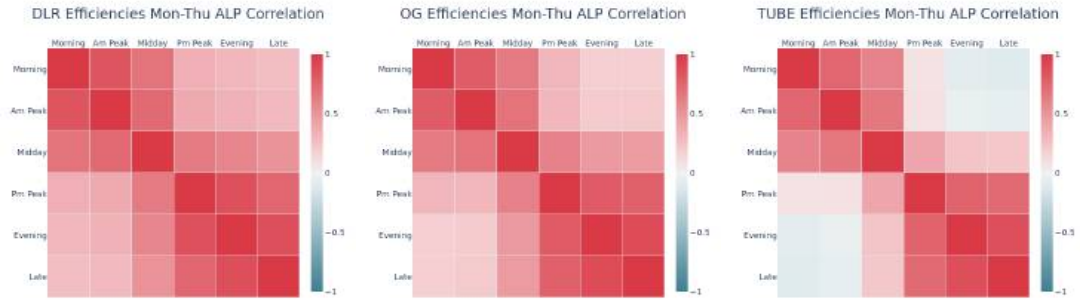

FIGURE 7.1: Traffic ranks heatmap across the periods.

The Heatmap figures for comparing traffic periods of DLR, Overground, and Tube systems concerning all days are shown in Appendix A, Sections A.9.

The Tube system exhibits the strongest inverse correlation for weekdays. This correlation can be attributed to the time overlap with working days, leading to the fair assumption that the Tube is primarily used by commuters traveling from home to work and vice versa. Consequently, inbound and outbound edges are deemed most "efficient" during the morning and evening periods, respectively. Below are plots comparing the morning and evening period backbone efficiencies for the Tube system.

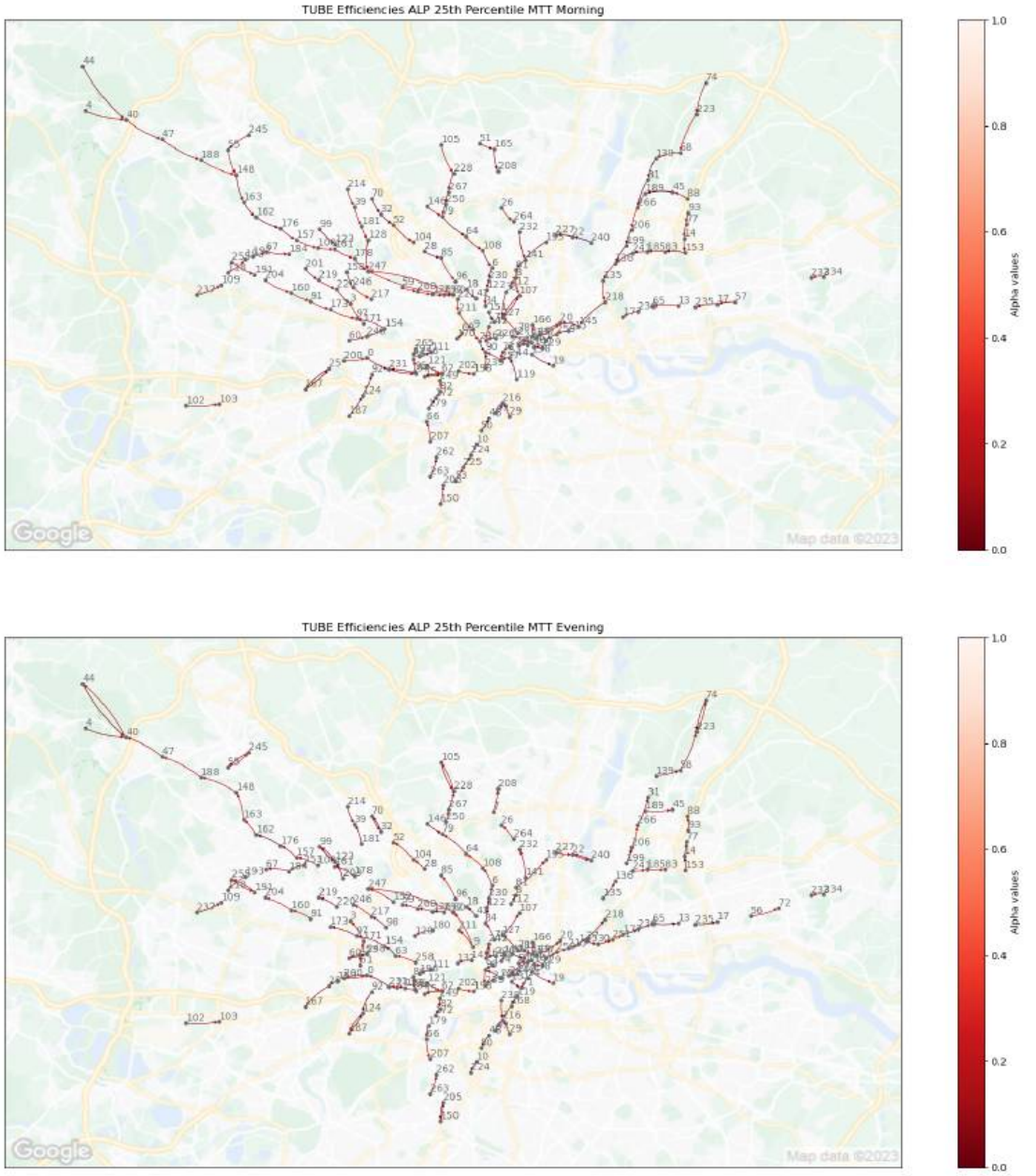

FIGURE 7.2: Tube efficiency 25th percentile backbone comparison.

The edges presumed to be the most efficient within the Tube system are evenly distributed across the London metropolitan area, suggesting no regional discrepancy in usage or the service provided. And similarly to the traffic analysis, there is a strong correlation across the days

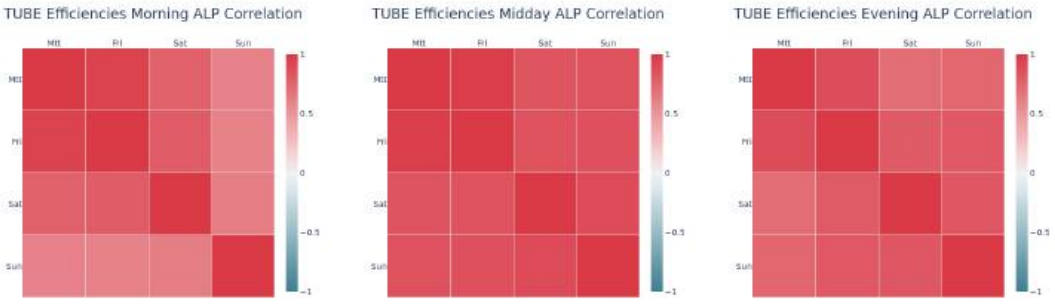

FIGURE 7.3: Tube efficiencies ranks heatmap across the days.

The Heatmap figures for comparing traffic days of DLR, Overground, and Tube systems concerning all days are shown in Appendix A, Sections A.24.

The disparity in edge efficiency ranks varies across days and periods, reflecting the dynamic nature of traffic within the transportation systems. This dynamic traffic pattern influences the utility of the systems, requiring operational adjustments to align with traffic dynamics and ensure optimal service provision relative to infrastructure capacity.

The backbone map covering the entire London metropolitan area suggests that operational guidelines, such as train speed and frequency, are proportionate to traffic volume. This implies that Transport for London (TFL) management does not prioritize allocating more trains of higher speeds to certain areas at the expense of others. However, this is only a preliminary analysis and further investigation is warranted, considering various granularities such as aggregation by time, areas, and more suitable indicators.

7.2 Robustness

The Robustness indicator, derived from the shortest-path algorithm, was utilized to weigh the edges. In determining the optimal path, the algorithm prioritizes time over distance. It's important to note that the model assumes a uniform speed value for each edge, resulting in no variation in the replacement path in case of disruption. However, the Edit distance incorporates traffic variations. As it is displayed below, the tube system presents a less strong correlation across the periods due to its strongest traffic variation.

Initially, the comparison will focus on how the two backbone extraction methodologies rank the edges' robustness.

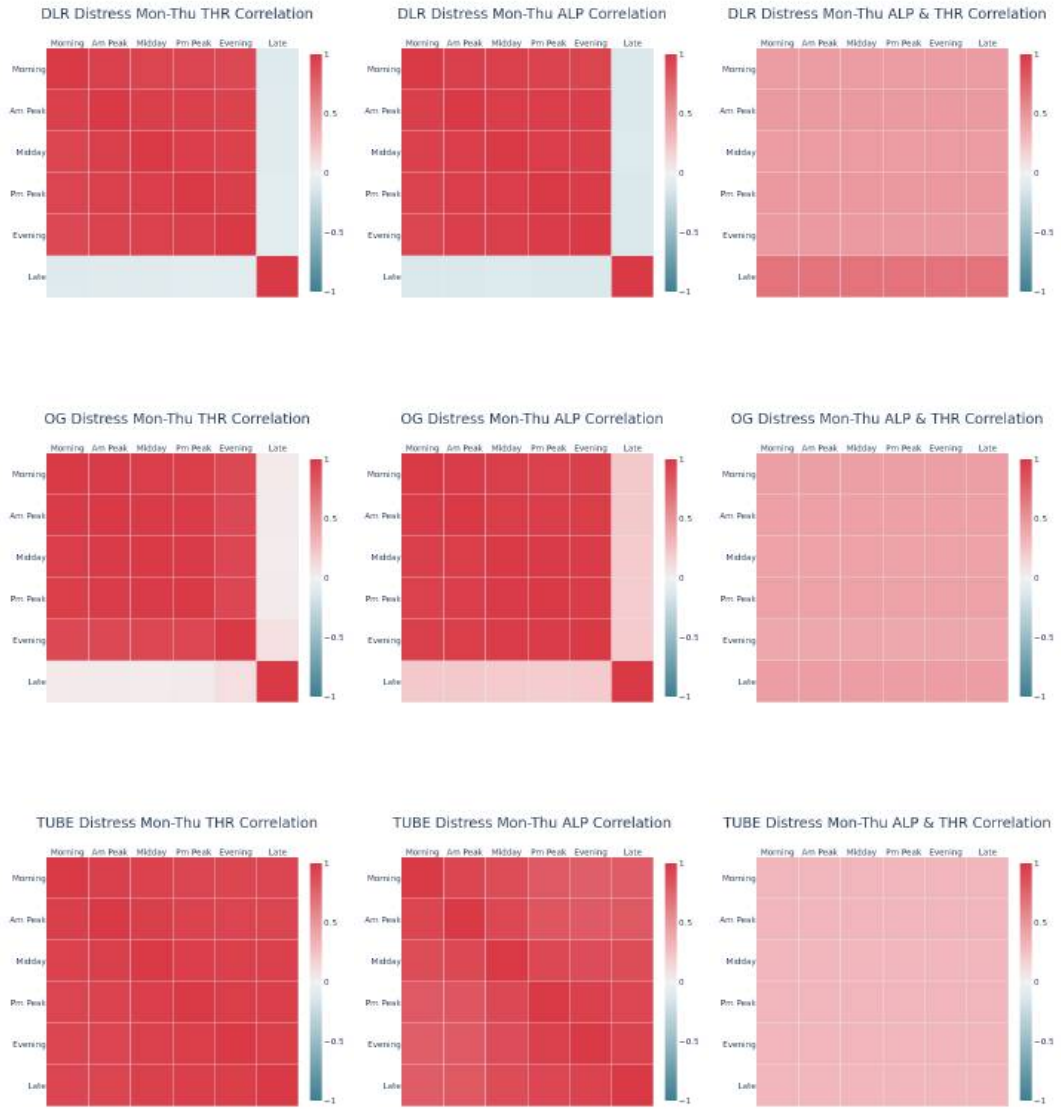

FIGURE 7.4: Distress ranks heatmap across the methods.

The Heatmap figures for comparing distress periods[1] of DLR, Overground, and Tube systems concerning all days are shown in Appendix A, Sections A.11.

Different from the Traffic model, It's challenging to discern a clear advantage between the backbone extraction methodologies for the Robustness model. Unlike with traffic, where the disparity method could capture traffic inversion indicated by negative correlation, there isn't a significant contrast between period ranks for robustness. This could be attributed to the fact that the indicators consider network structure, whereas the disparity filtering may not contribute much structural information.

[1]The Late period must not be considered for DLR and Overground systems, see section 4.7

Therefore, at this point, it's not possible to imply a superior grasp of the indicator, but rather that both methodologies offer different perspectives on edge robustness evaluation. The third column shows a moderate positive correlation between their evaluations.

Considering the Tube system, the image below illustrates the rank correlation comparison across periods.

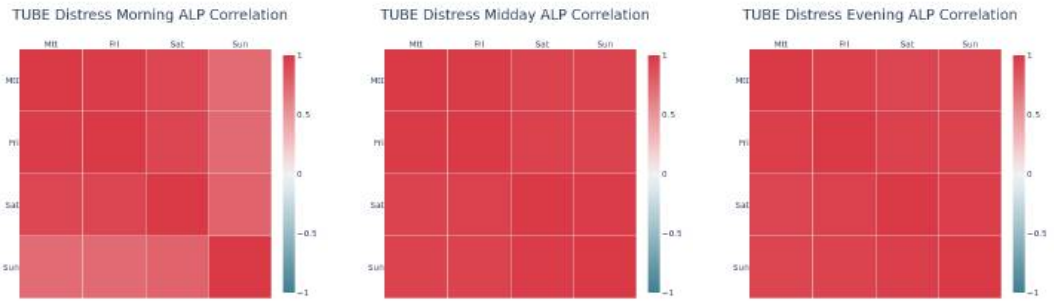

FIGURE 7.5: Tube distress ranks heatmap across the periods.

The Heatmap figures for comparing distress days of DLR, Overground, and Tube systems concerning all days are shown in Appendix A, Sections A.12.

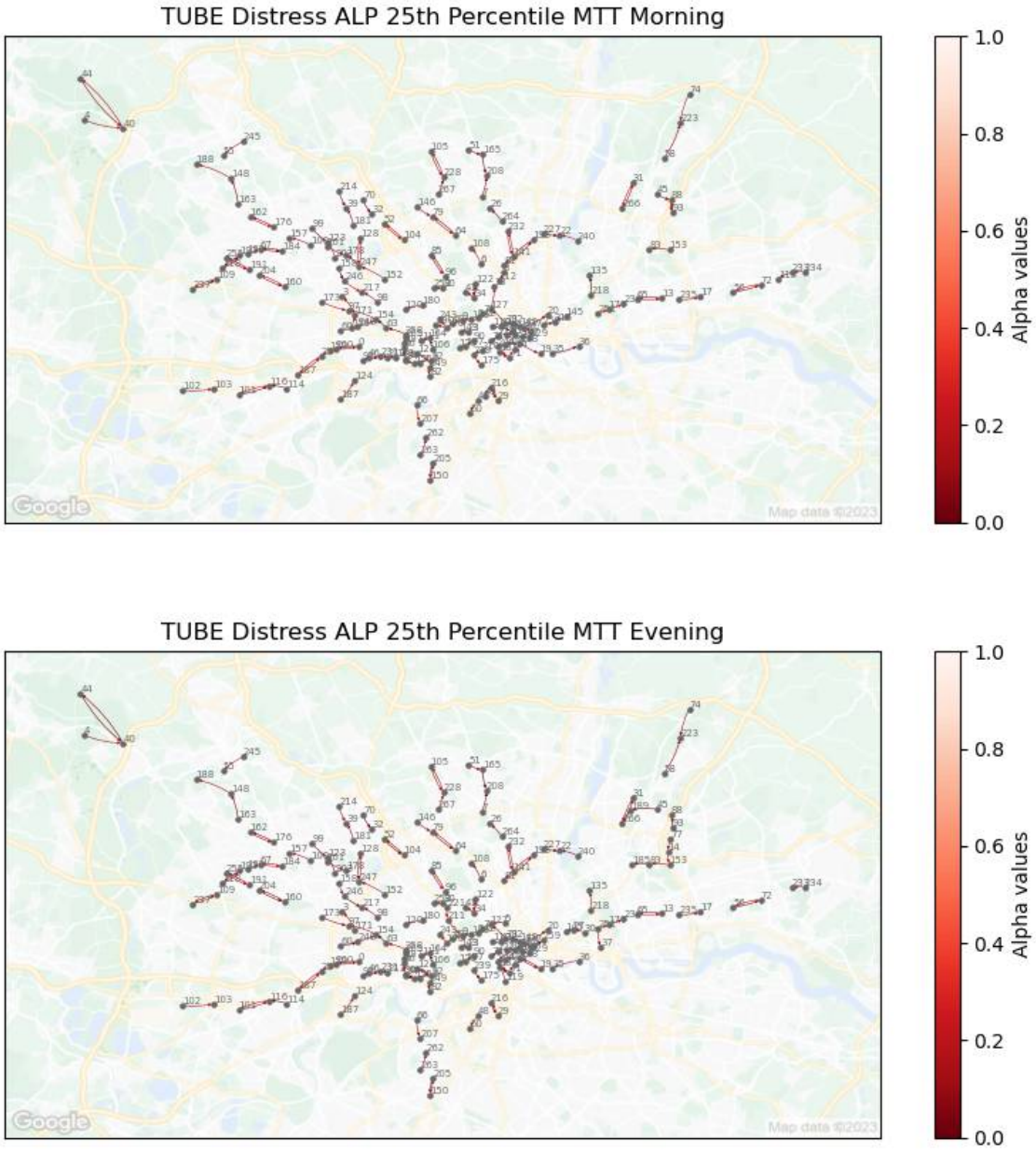

FIGURE 7.6: Tube efficiency 25th percentile backbone comparison.

Although the robustness backbone shows some similarity to the efficiency backbone, a noticeable disconnection exists between the stations. The most vulnerable edges are those with less redundancy, which, following the nature of a transport network, are located in peripheral areas. The network ramification increases in central locations, enhancing the edges' robustness. Another significant difference is the invariance of the edge's direction. The distress indicator, based on the shortest path algorithm, embeds the network structure but undermines traffic variation.

On the other hand, the efficiency indicator is heavily driven by traffic, which attributes the network structure aspect of the backbone to the disparity filtering methodology.

7.3 Efficiency and Robustness Comparison

The heatmaps below compare the ranking from the distress and efficiency models for each transport system.

FIGURE 7.7: Efficiencies and distress ranks heatmap across systems from Monday to Thursday.

The Heatmap figures for comparing efficiencies and distress rankings for DLR, Overground, and Tube systems across all days are shown in Appendix A, Sections A.13.

While the Overground and Tube systems exhibit a low negative correlation, the DLR system shows a low positive correlation. This can be attributed to the fact that the DLR system has a significantly lower number of edges and presents a clear central core through which most journeys pass. The comparison of the 25th percentile backbones further emphasizes this positive similarity.

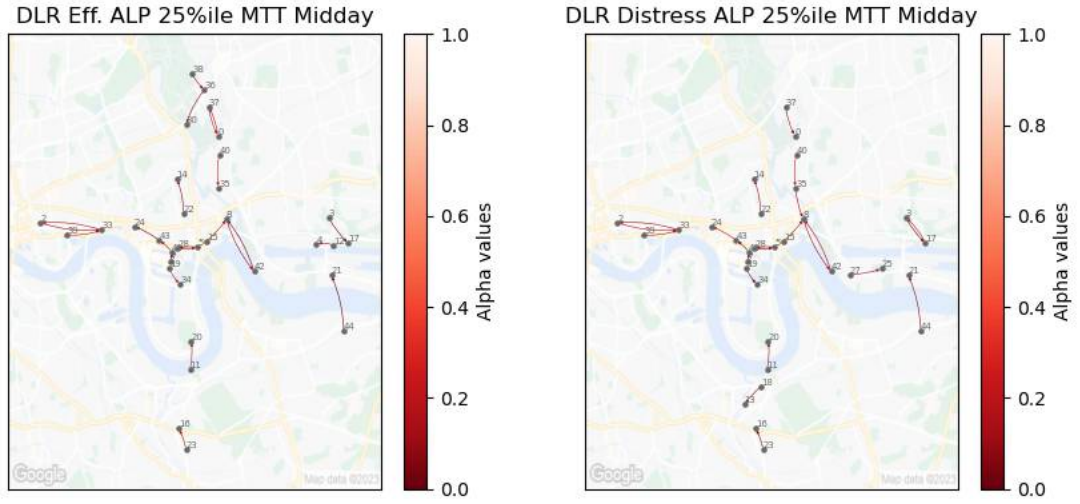

FIGURE 7.8: Relevance comparison for edges speed.

The Heatmap figures for comparing efficiencies backbones of Overground, and Tube systems across all days are shown in Appendix A, Sections A.14.

In comparing robustness and efficiency models, it becomes evident that they offer orthogonal perspectives of the network. Both methods come with their own set of assumptions and limitations, which are crucial to consider during analysis. What's interesting is how these two perspectives complement each other. Each extraction method and indicator aims to measure different aspects of the transport network, such as equality, efficiency, resource allocation, and resilience against failures.

Although it's possible to merge the information to create a single rank, that's not the intention here. Instead, the goal is to highlight the diverse approaches available for specific management practices, ultimately contributing to the overall optimization of network management. However, certain edges can still be pinpointed that hold significance in the daily lives of London users, shedding light on both robustness and efficiency scenarios.

Of course, there's always room for improvement to achieve a deeper and more realistic analysis. This could entail refining indicators with more complex metrics, enhancing modeling by incorporating additional data and processing power or adjusting methodologies, such as fine-tuning disparity filtering to better suit transportation-specific needs.

The complete ranks can be found in Appendix B, section **??**, provided in CSV format for easy reference.

Chapter 8

Following steps

The integration of transport systems is widely recognized as crucial for achieving a seamless and efficient mobility experience. However, accurately estimating connections and transitions between modes of transport presents significant challenges. To overcome these obstacles and potentially pave the way for substantial advancements, the implementation of an integrated system that faithfully represents real-life scenarios is deemed imperative.

To optimize transportation systems, the next logical step would involve the practical application of advanced AI and machine learning techniques. This structured approach could comprise three integral parts, each potentially contributing to enhancing system efficiency and responsiveness.

In an initial phase, it is conceivable that machine learning algorithms could be harnessed to predict passenger origins and destinations. Advanced mathematical techniques, potentially informed by an understanding of the network structure, might facilitate effective modeling of transport systems. By analyzing potentially vast amounts of transportation data, these algorithms could theoretically predict passenger demand, optimize resource allocation, and improve scheduling decisions. Enhanced methods such as inferring rail trip OD matrices could provide fundamental information for planning and operations analysis, potentially improving the accuracy and reliability of modeling efforts.

Moreover, our approach could extend beyond data analysis to potentially map passenger profiles using available information to discern their preferences and favored routes. By potentially understanding travel patterns and ticket sales, we could gain insights into demand dynamics, potentially enabling us to tailor transportation services more effectively to meet passenger needs. It would be crucial to uphold stringent measures to

ensure the anonymity and confidentiality of passenger data, adhering to strict ethical standards and regulatory guidelines.

The second part could involve disruption simulation and route optimization. Through meticulous scenario analysis, we could theoretically simulate various disruption scenarios to understand their potential impact on network efficiency. Sophisticated algorithms could be employed to compute alternative routes and conduct stress analysis to ensure seamless adaptation during disruptions, potentially minimizing passenger inconvenience and preserving optimal network functionality.

Finally, reinforcement learning could theoretically be utilized to guide operational decisions within the transportation domain. An AI agent could be tasked with formulating operational guidelines for timetables, maintenance schedules, and infrastructure improvements. Operating within predefined budget constraints, the agent could aim to optimize network performance by reducing distress, enhancing global efficiency, and curbing operational costs. Continuous learning and adaptive strategies could potentially ensure the agent's efficacy in real-world scenarios.

These strategic steps, potentially informed by advanced AI methodologies, could potentially propel transportation systems towards greater safety, efficiency, and responsiveness, representing a possible pivotal advancement in transportation optimization and potentially paving the way for a future characterized by seamless mobility and enhanced societal well-being.

Chapter 9

Conlcusion

Is it feasible to manage the transportation system optimally to meet the needs of passengers? This study underscores the power of the disparity filter method in addressing this inquiry. However, its effectiveness hinges on its customization to the specific complexities of each transportation network. As discussed in the previous chapter, achieving such customization involves leveraging multiple disciplinary tools and historical data, while also accounting for dynamic variables such as traffic forecasts and urban dynamics.

Undoubtedly, every transportation system in the world was created based on momentary demand, but the real challenge lies in understanding and optimizing these systems to evolve with future demands. An interesting question arises: how can we tailor the disparity filter method to better address the nuances of each system and provide meaningful insights into traffic patterns and network robustness? Emerging research has introduced alternative backbone extraction methodologies, juxtaposed with the established approach of the disparity filter.

The Global Statistical Significance (GloSS) filter [6] stands out for its ability to identify relevant connections while preserving both the weight distribution and the full topological structure of the network. Unlike a specific probability density function, the GloSS method utilizes the observed weight distribution of the network under study. By constructing a null model with randomly assigned weights from the observed distribution, statistical relevance is calculated using a Bayesian approach.

An alternative to local and global methods is an adaptive extraction method [7], which is a designed backbone extraction technique accounting for both global and local topological structures of networks. This model closely resembles the disparity filtering method but employs the shortest paths involving the node rather than its strength to

measure involvement. Introducing a variable to regulate the impact of the node degree on its statistical relevance adds flexibility but may compromise analytical results as the backbone may be disproportionately influenced by local or global structure.

The true strength of the disparity filter lies in its adaptability to different operational premises within transportation systems. By customizing the method to suit the system characteristics, we can unlock its full potential in analyzing traffic flow and optimizing network efficiency. The null model is akin to prior knowledge in Bayesian statistics. The network's historical data could be used to define the probability density function to highlight the fluctuations in context with each analytical objective.

In conclusion, while the disparity filter method stands out as a powerful tool in transportation system analysis, its true effectiveness lies in its tailored application to each system's complexities, alongside the utilization of null models to provide meaningful context and insights. By embracing these principles, we can harness the full potential of the disparity filter method to optimize transportation systems and meet the evolving needs of passengers effectively.

Appendix A

A.1 Shorstest Path Exception

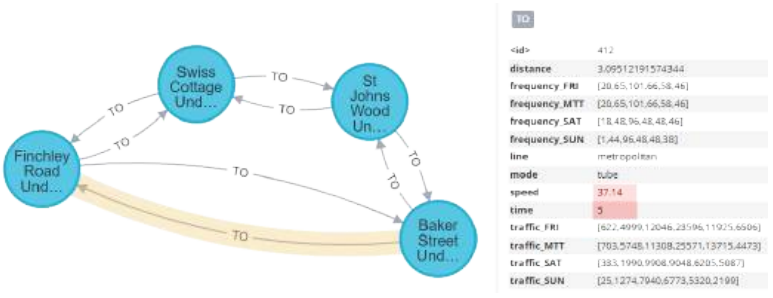

FIGURE A.1: Shortest path concerning number of stations.

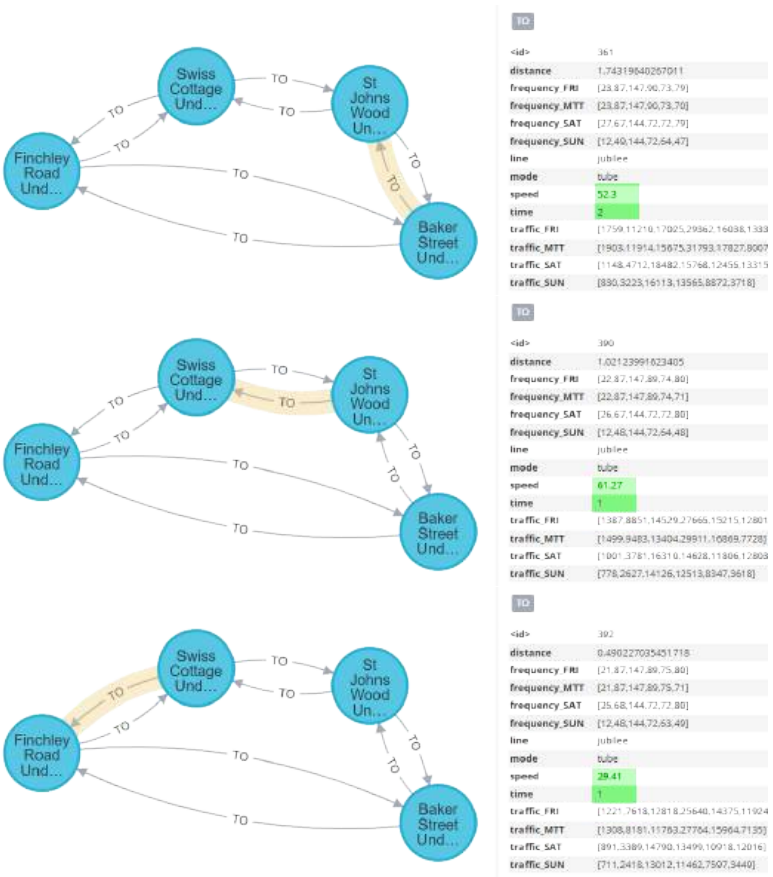

FIGURE A.2: Shortest path concerning time.

A.2 Backbone Extractions by Distance

A.2.1 Overground Distances

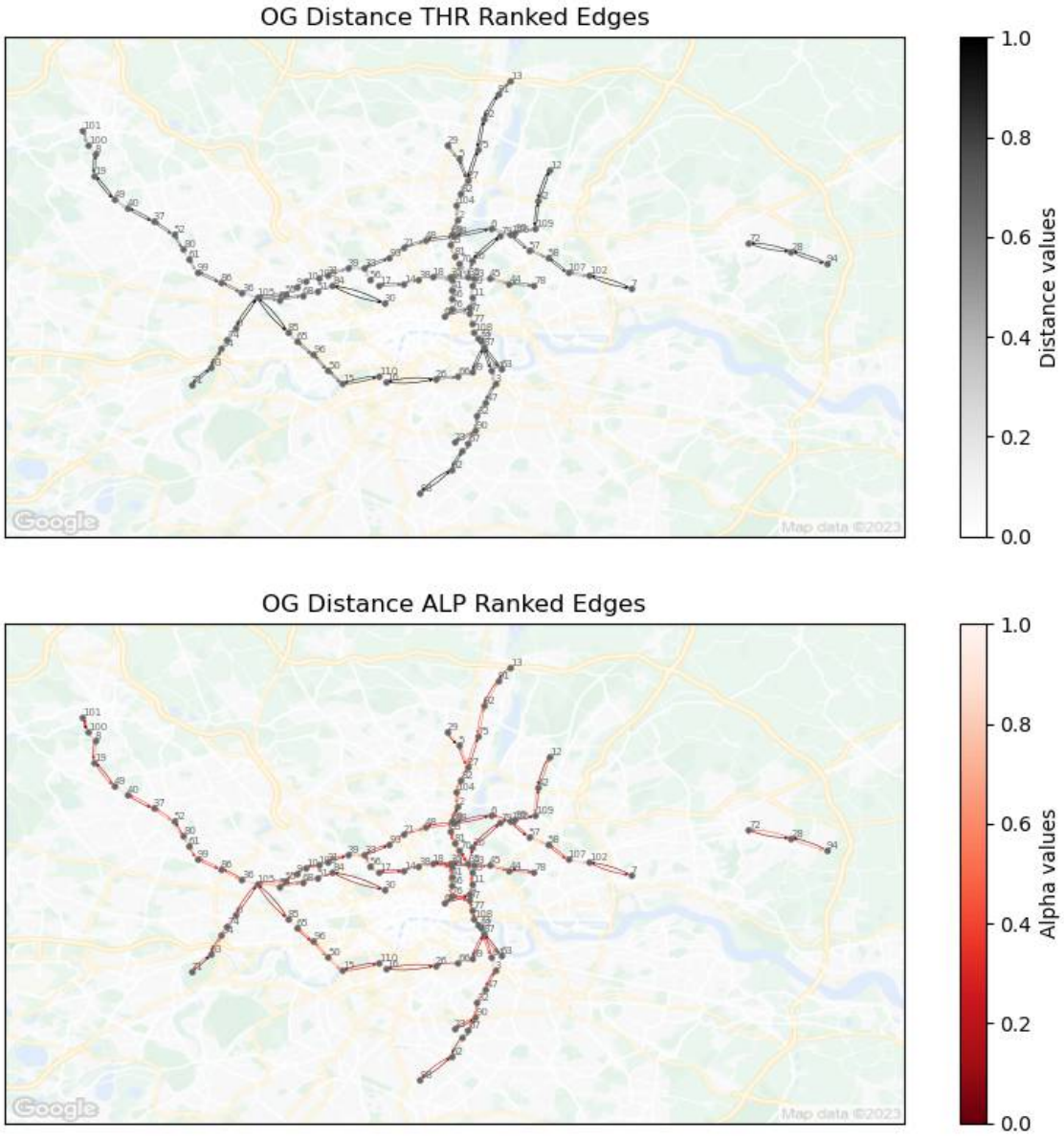

FIGURE A.3: Overground distance rankings.

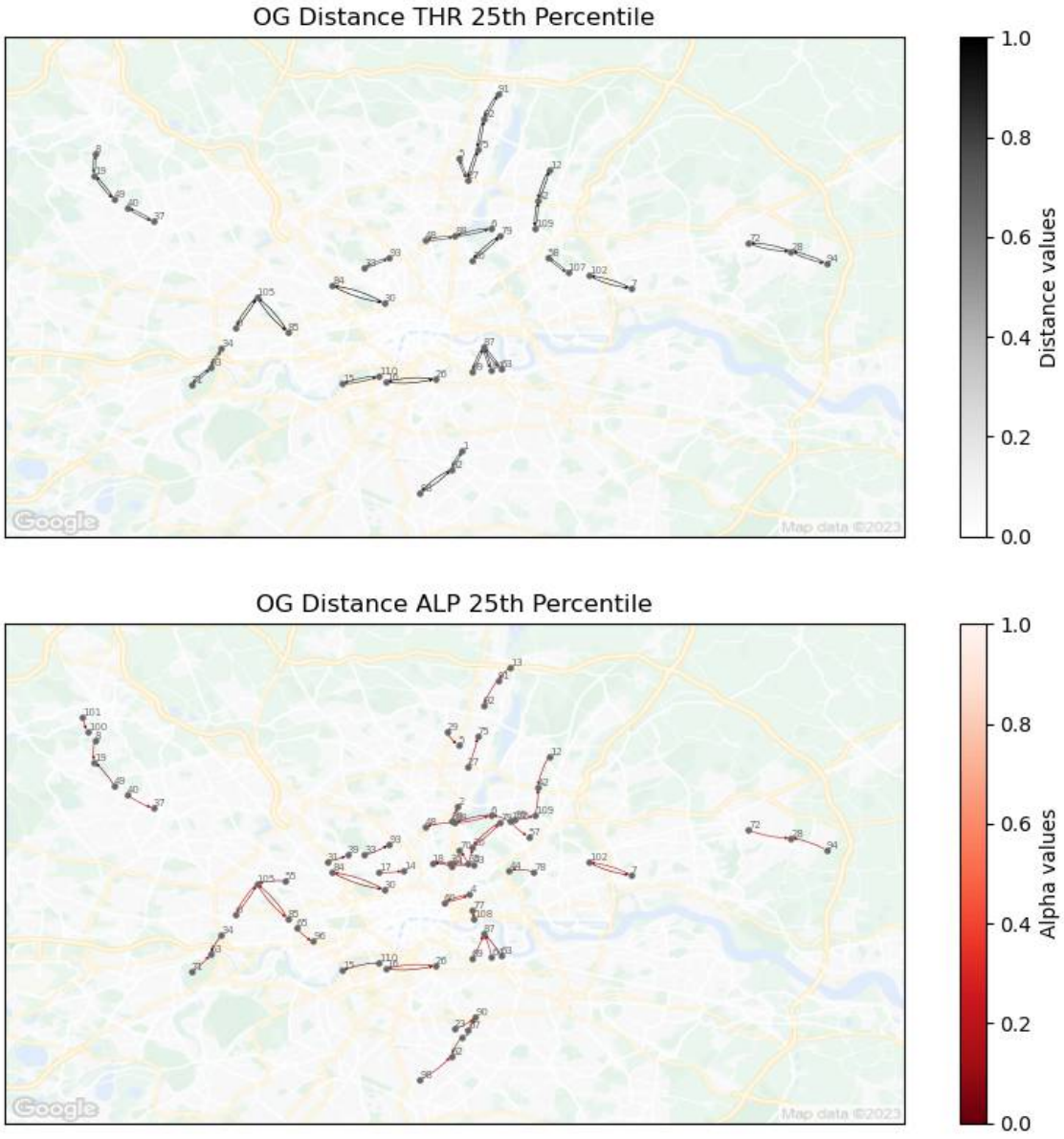

FIGURE A.4: Overground distance 25th percentile backbones.

A.2.2 Tube Distances

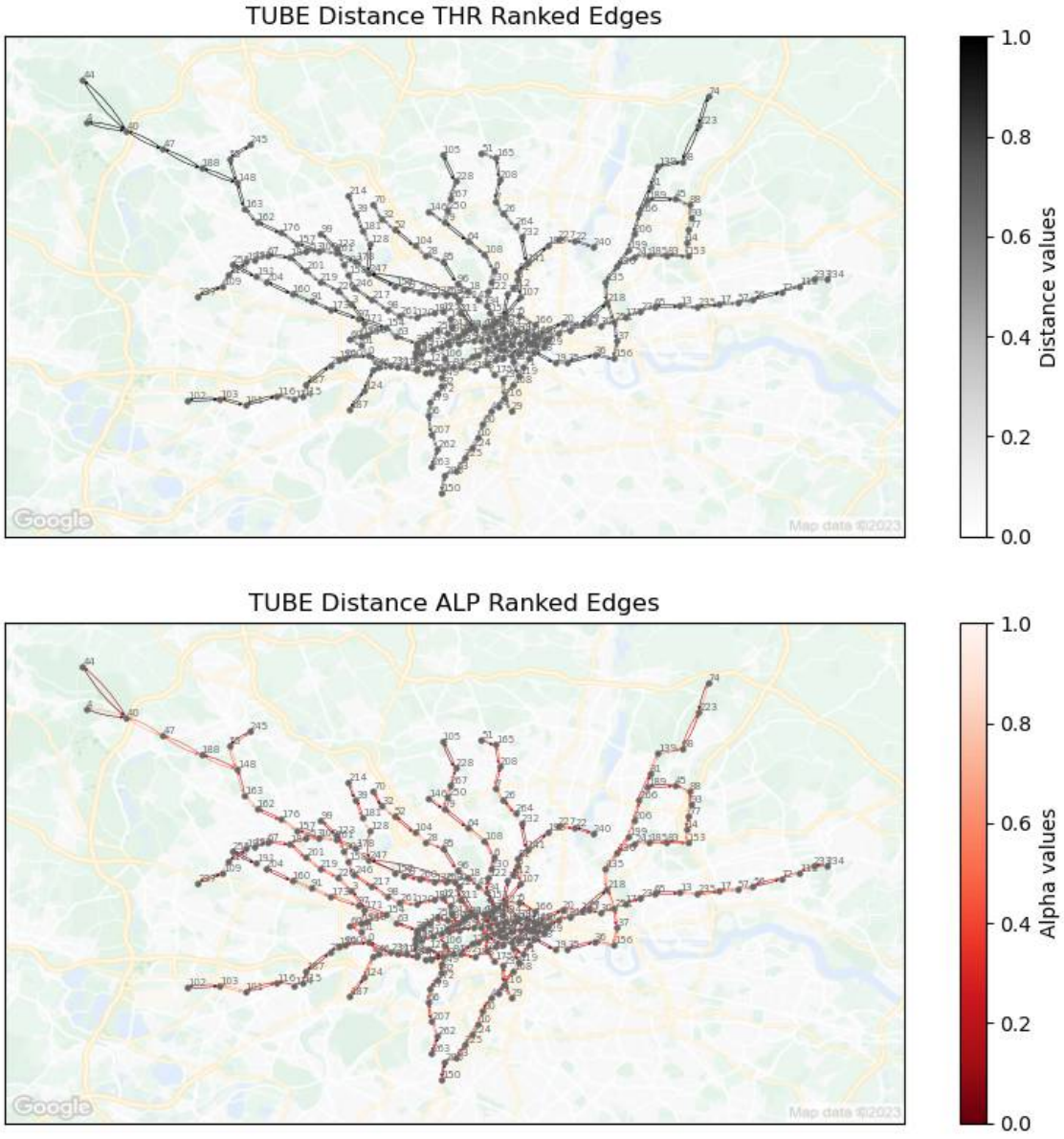

FIGURE A.5: Tube distance rankings.

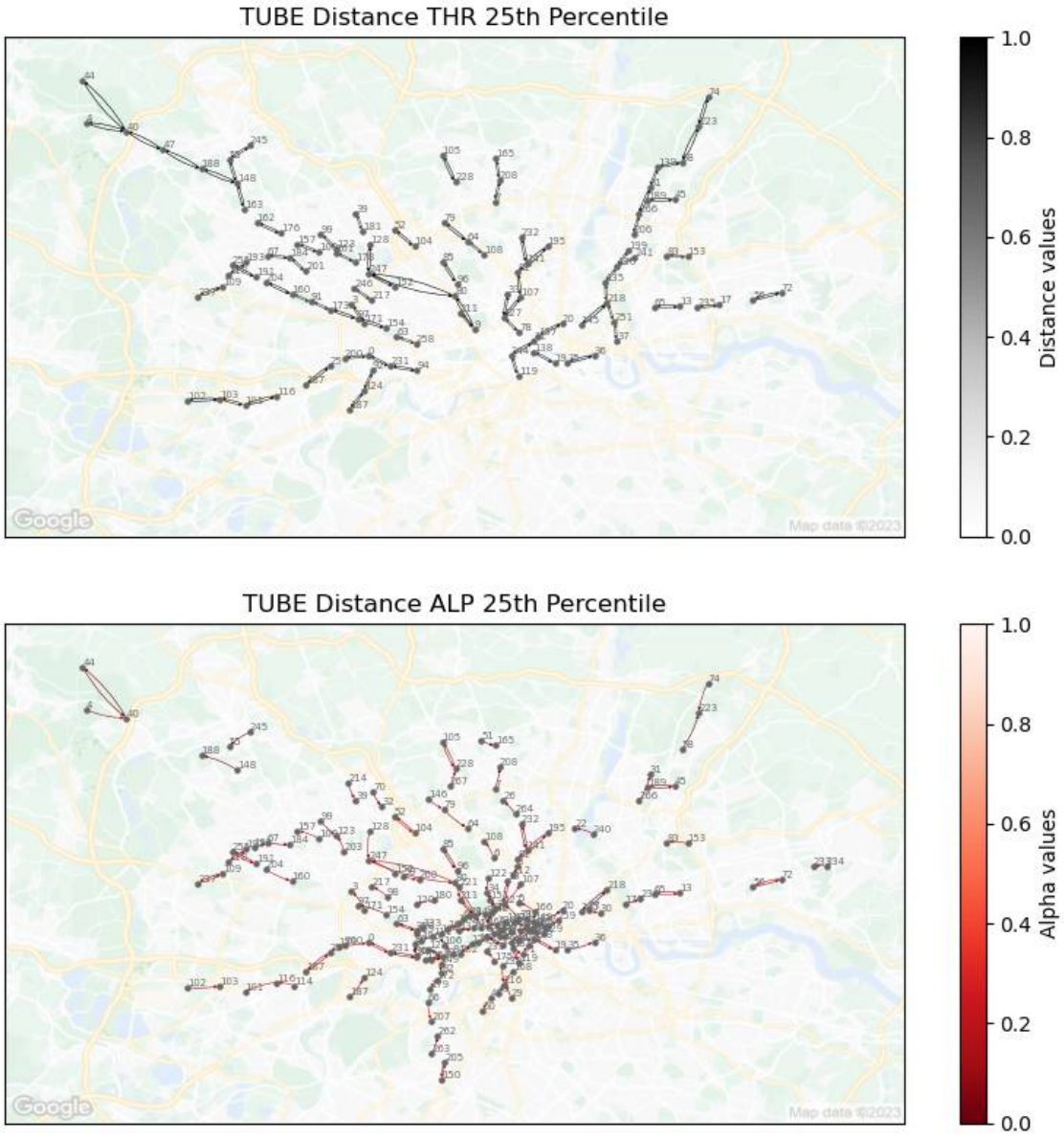

FIGURE A.6: Tube distance 25th percentile backbones.

A.3 Extraction Backbone by Speed

A.3.1 Overground Speeds

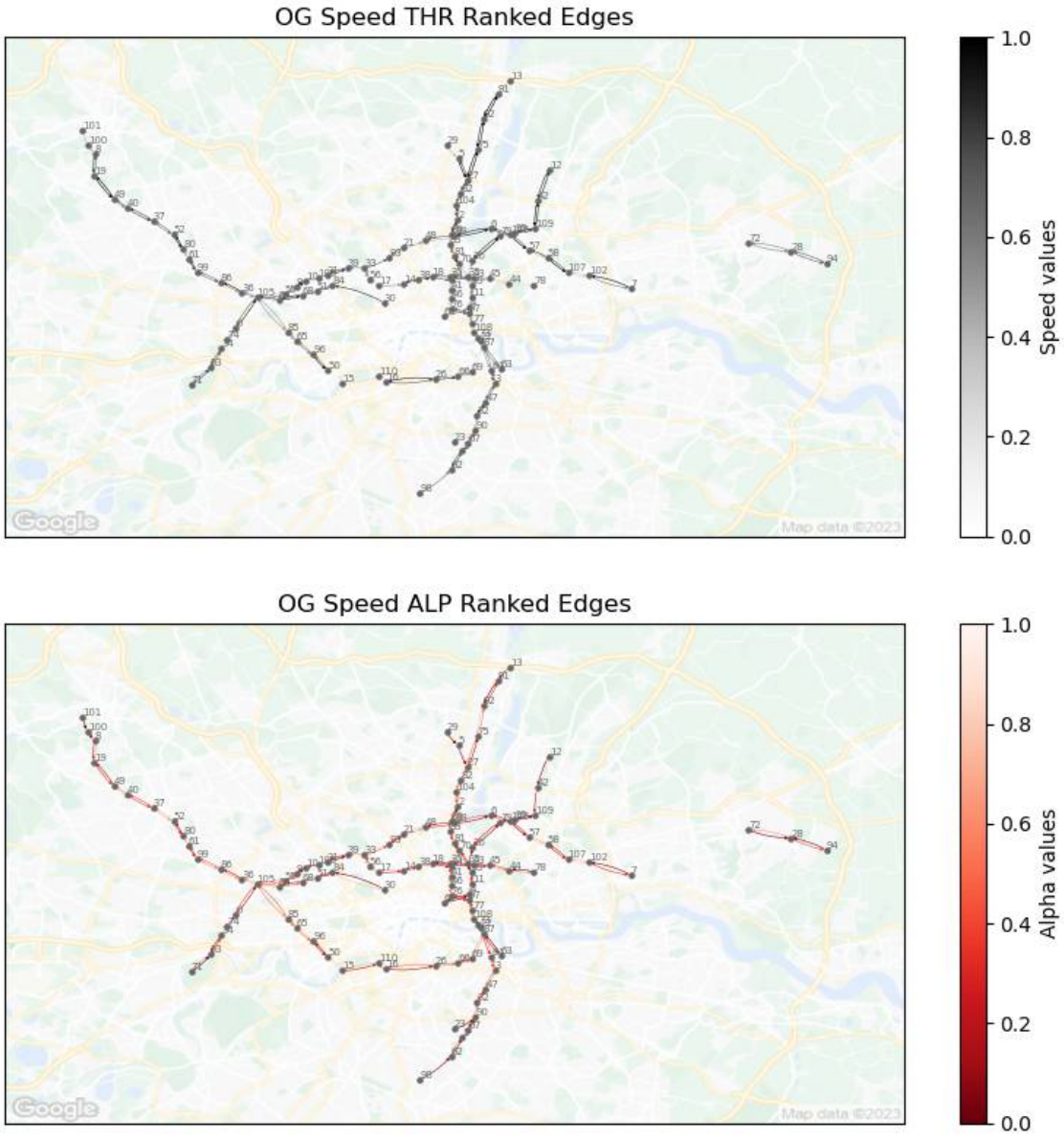

FIGURE A.7: Overground speed rankings.

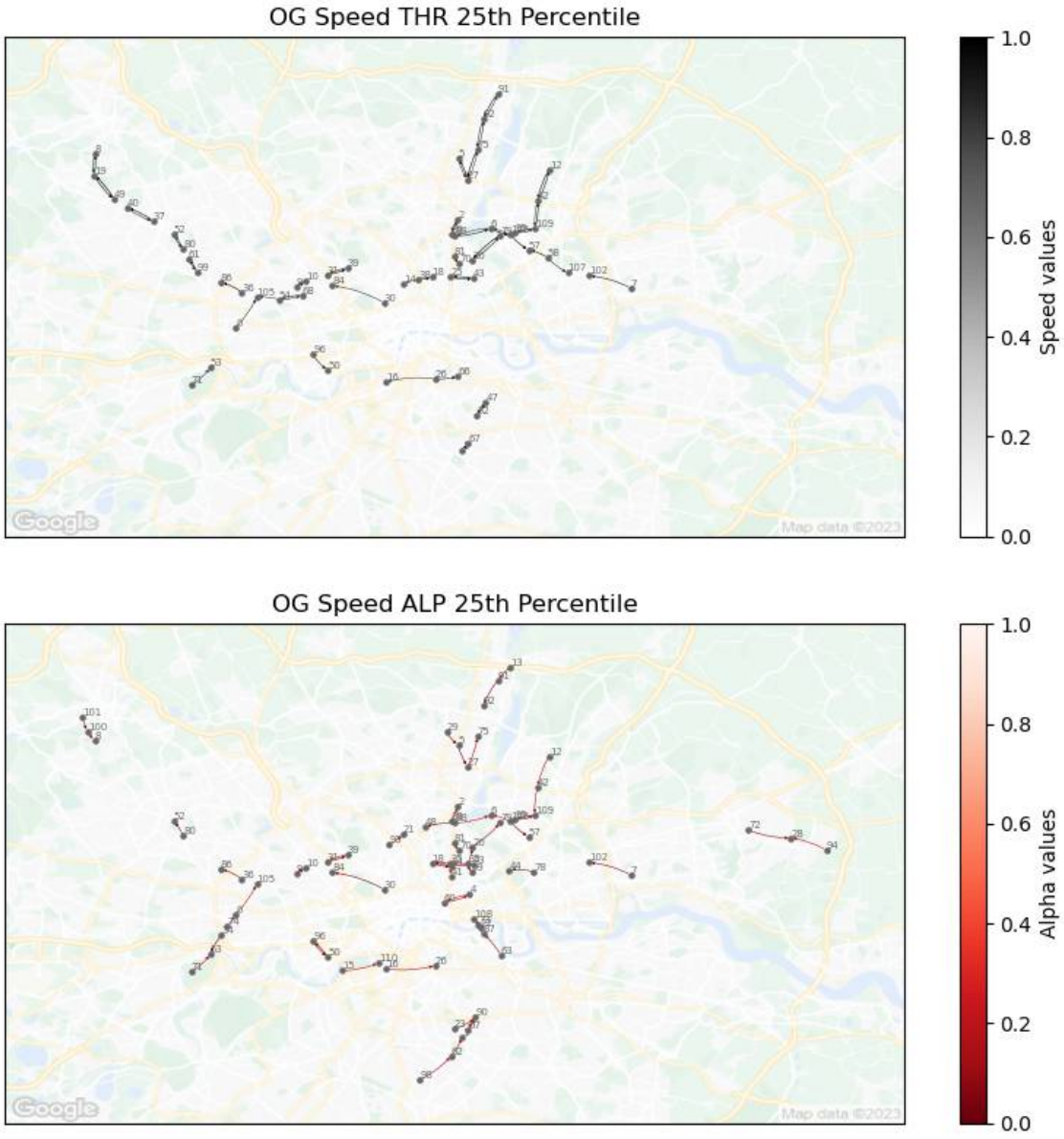

FIGURE A.8: Overground speed 25th percentile backbones.

A.3.2 Tube Speeds

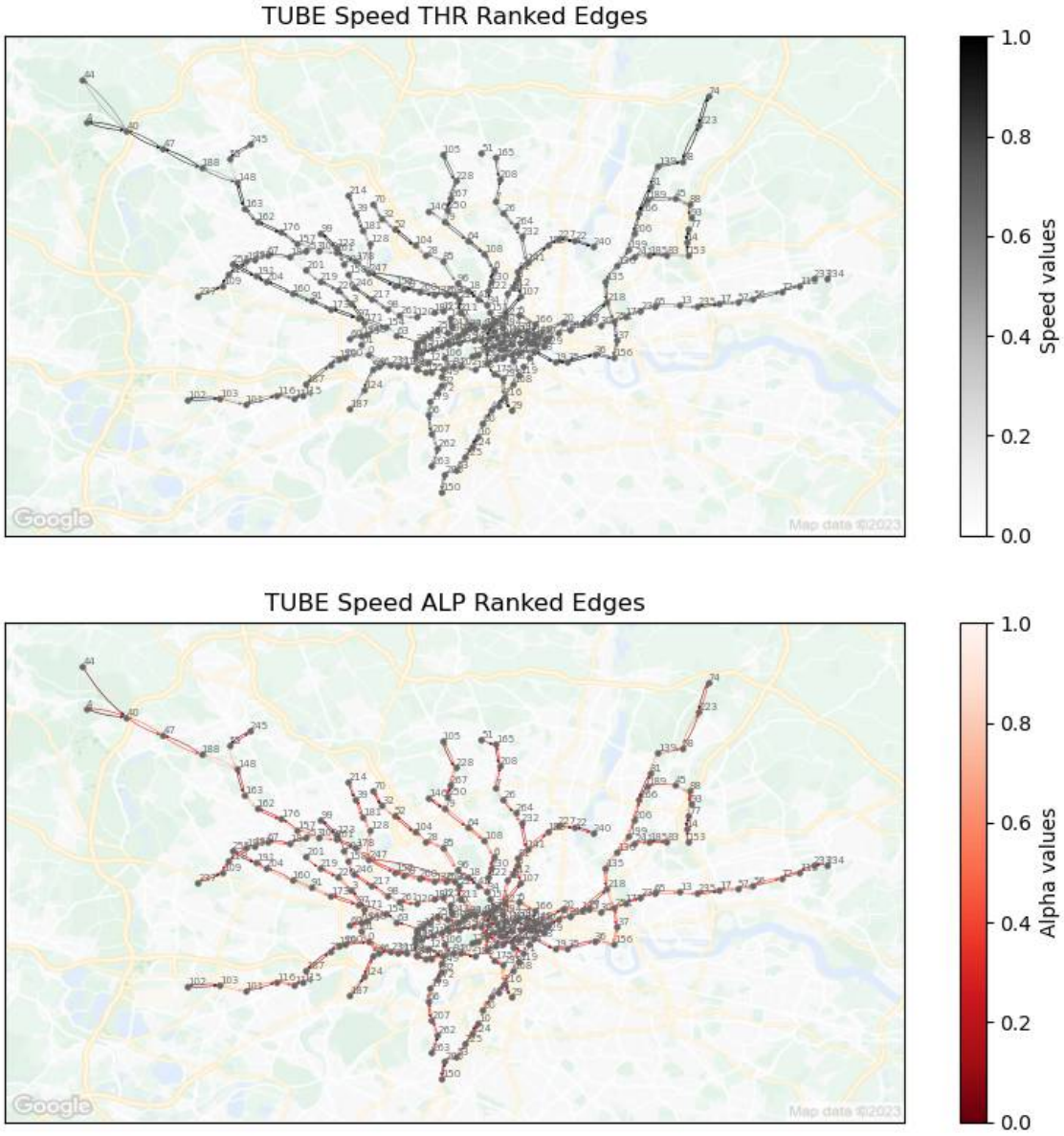

FIGURE A.9: Tube speed rankings.

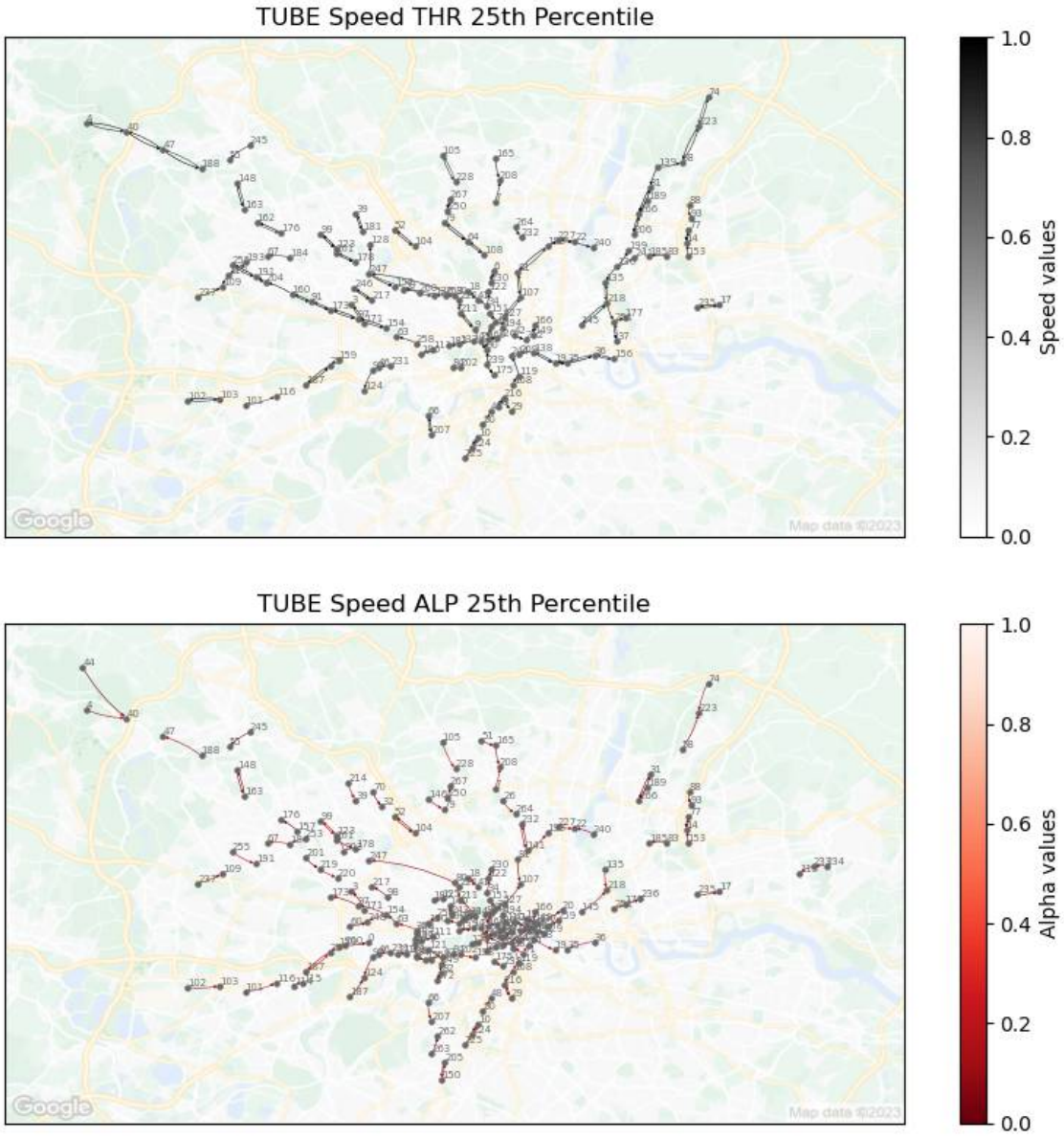

FIGURE A.10: Tube speed 25th percentile backbones.

A.4 Extraction Methods Comparison

A.4.1 Overground Traffic Backbones by Periods

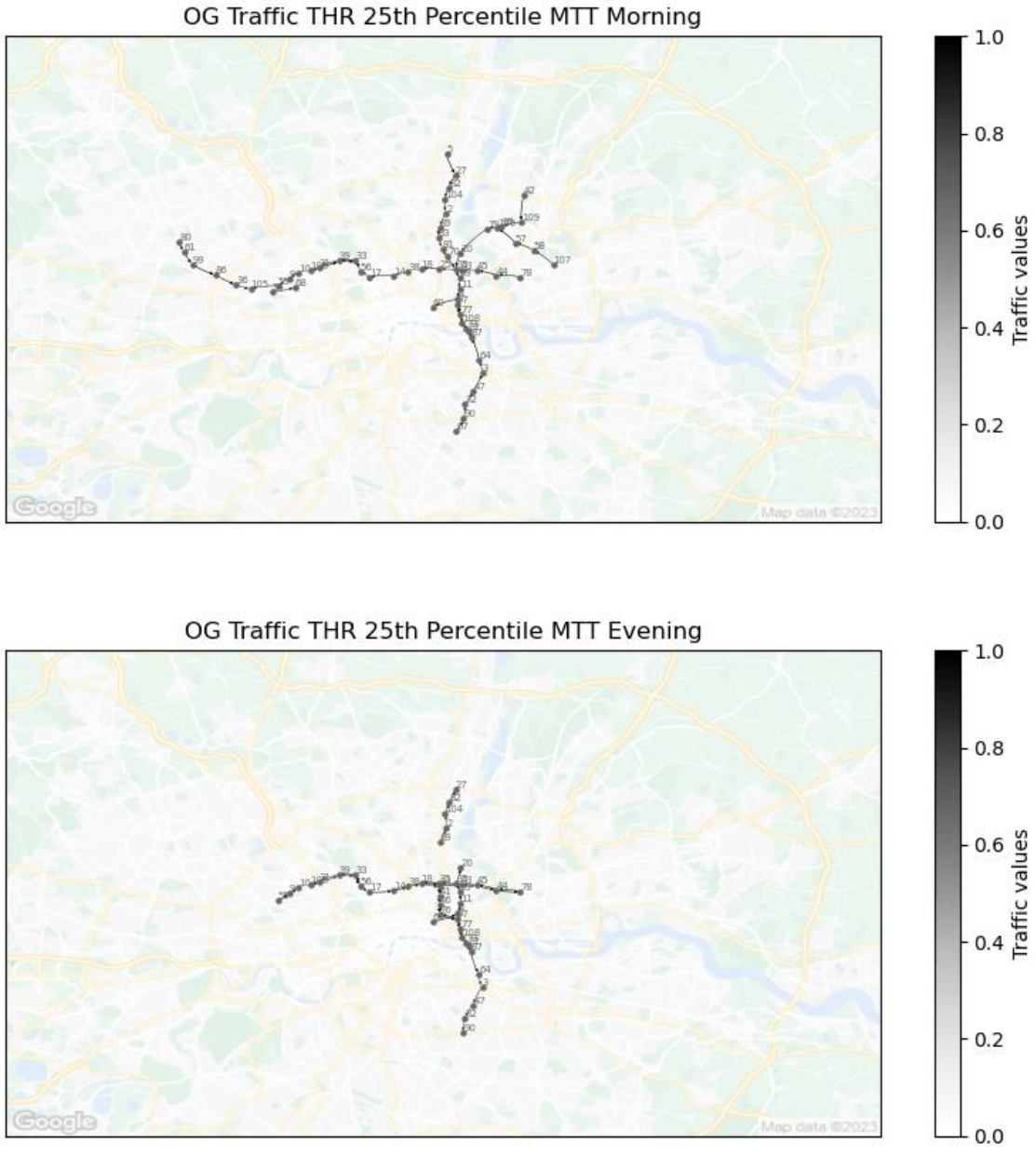

FIGURE A.11: Overground traffic 25th percentile thresholding backbones.

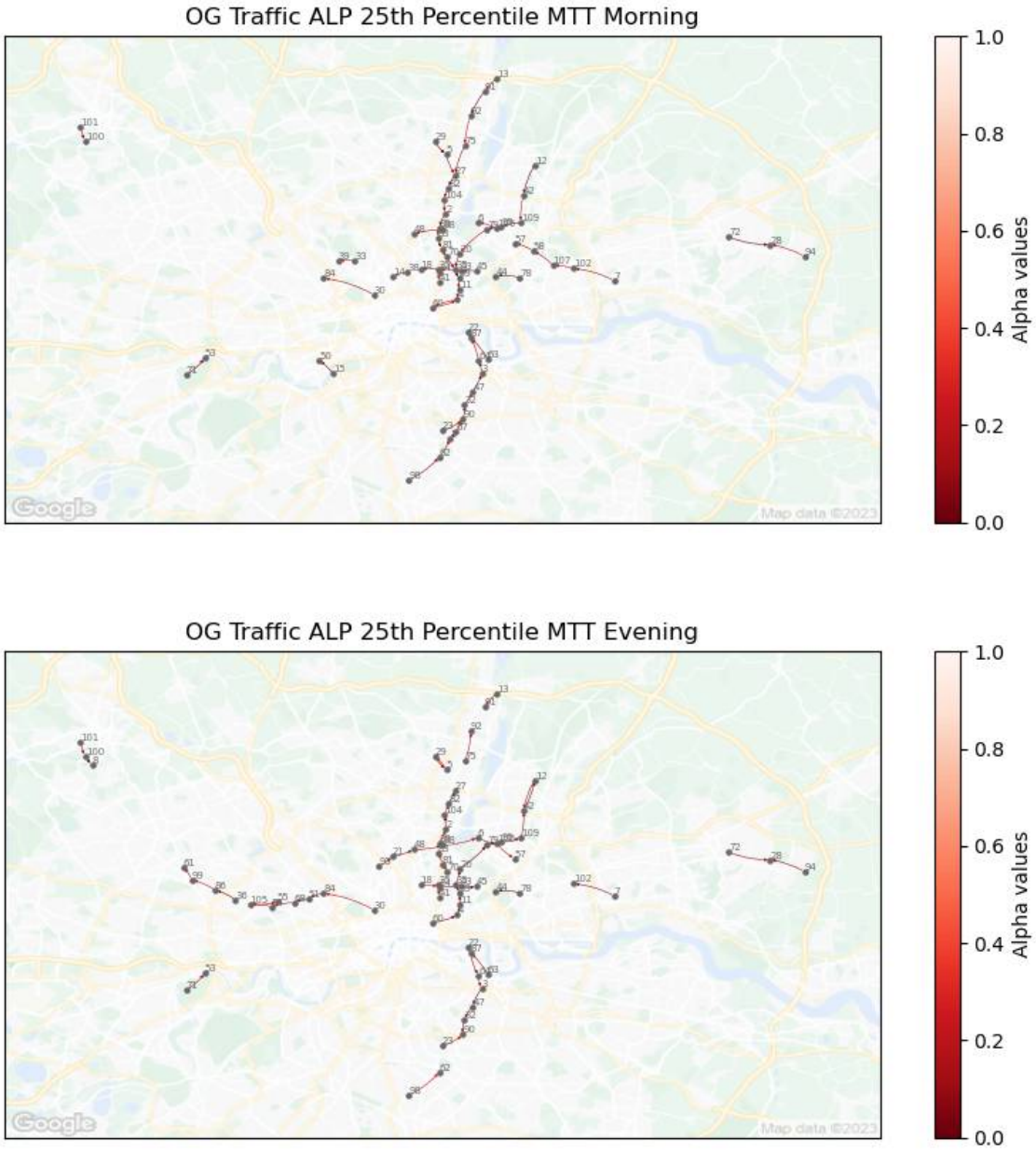

FIGURE A.12: Overground traffic 25th percentile disparity backbones.

A.4.2 Tube Traffic Backbones by Periods

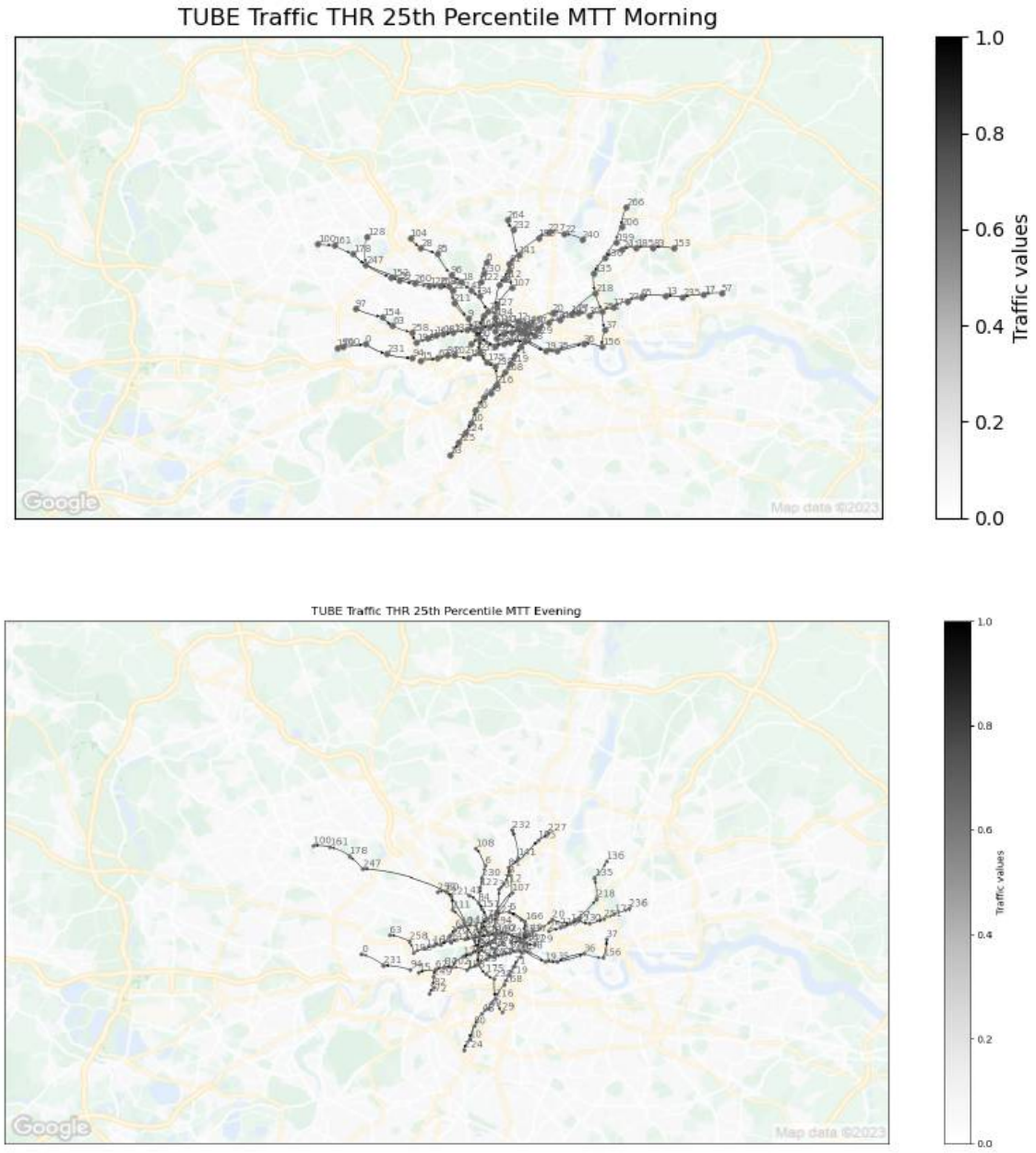

FIGURE A.13: Tube traffic 25th percentile thresholding backbones.

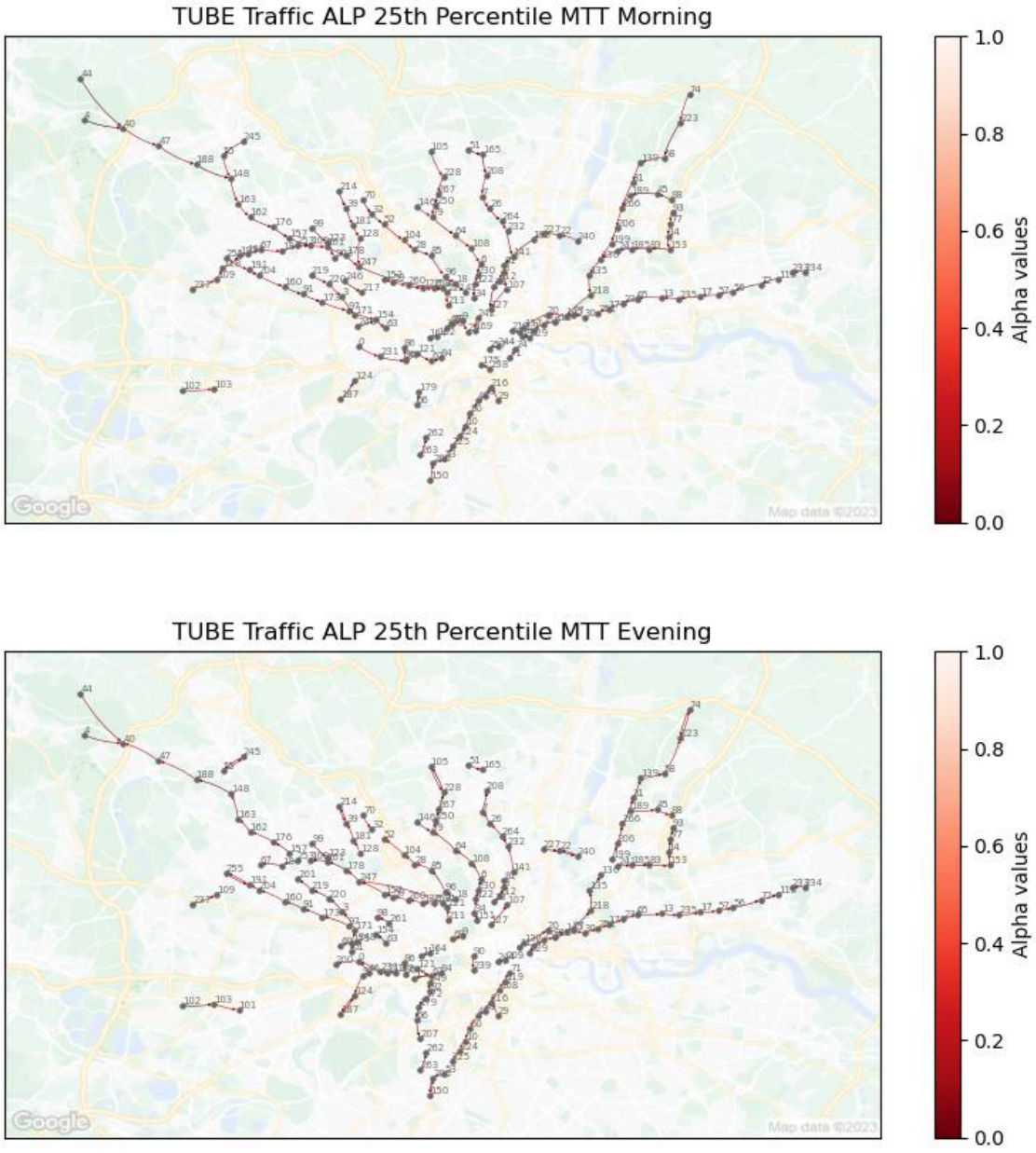

FIGURE A.14: Tube traffic 25th percentile disparity backbones.

A.5 Stations Entries and Exists

A.5.1 Overground Stations Entries and Exists for MTT

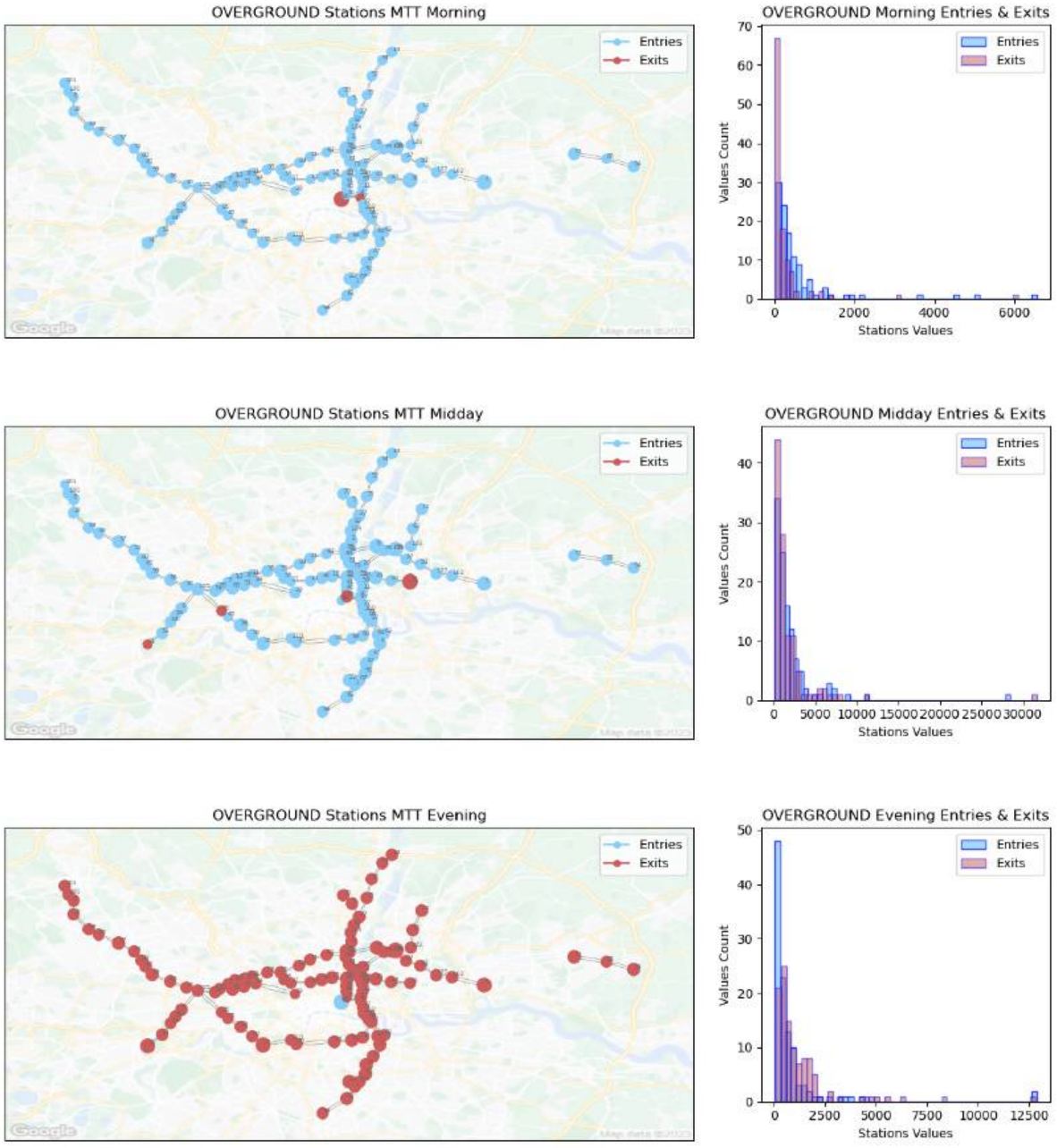

FIGURE A.15: Overground station entries & exist for morning, midday, and evening periods.

A.5.2 Tube Stations Entries and Exists for MTT

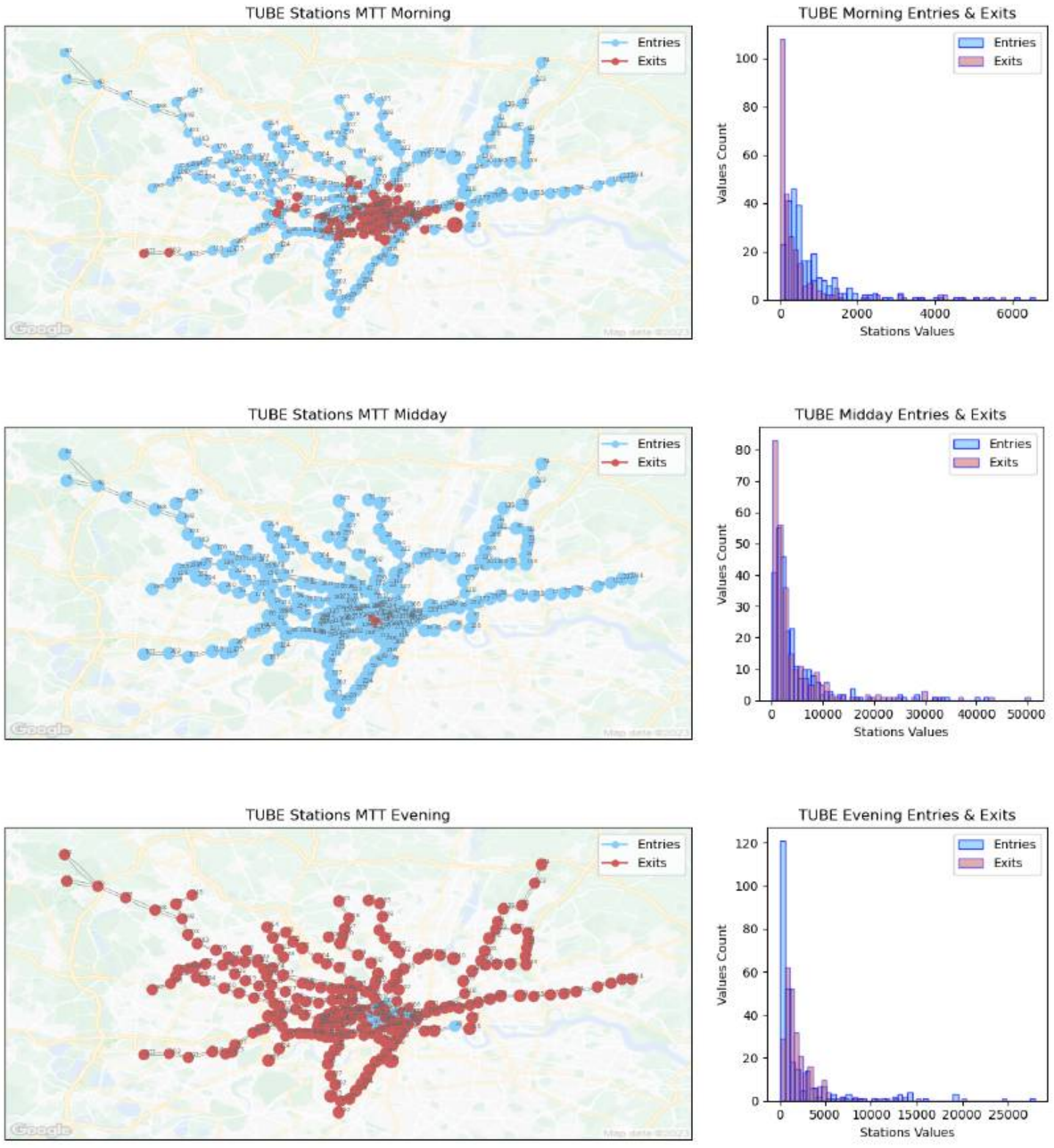

FIGURE A.16: Tube station entries & exist for morning, midday, and evening periods.

A.6 Extraction Backbone by Traffic

A.6.1 Overground Traffic by Days

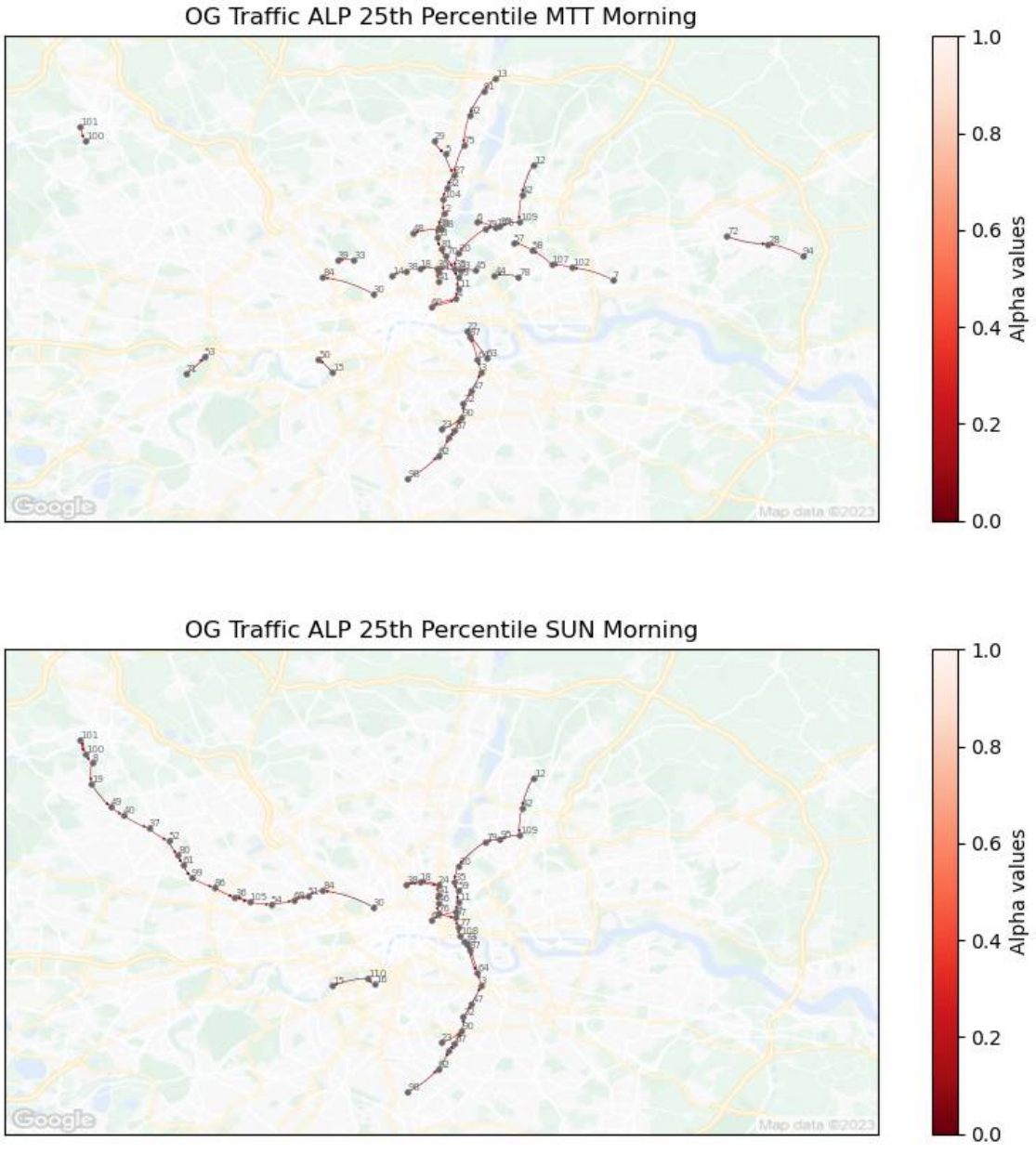

FIGURE A.17: Overground traffic 25th percentile thresholding backbones.

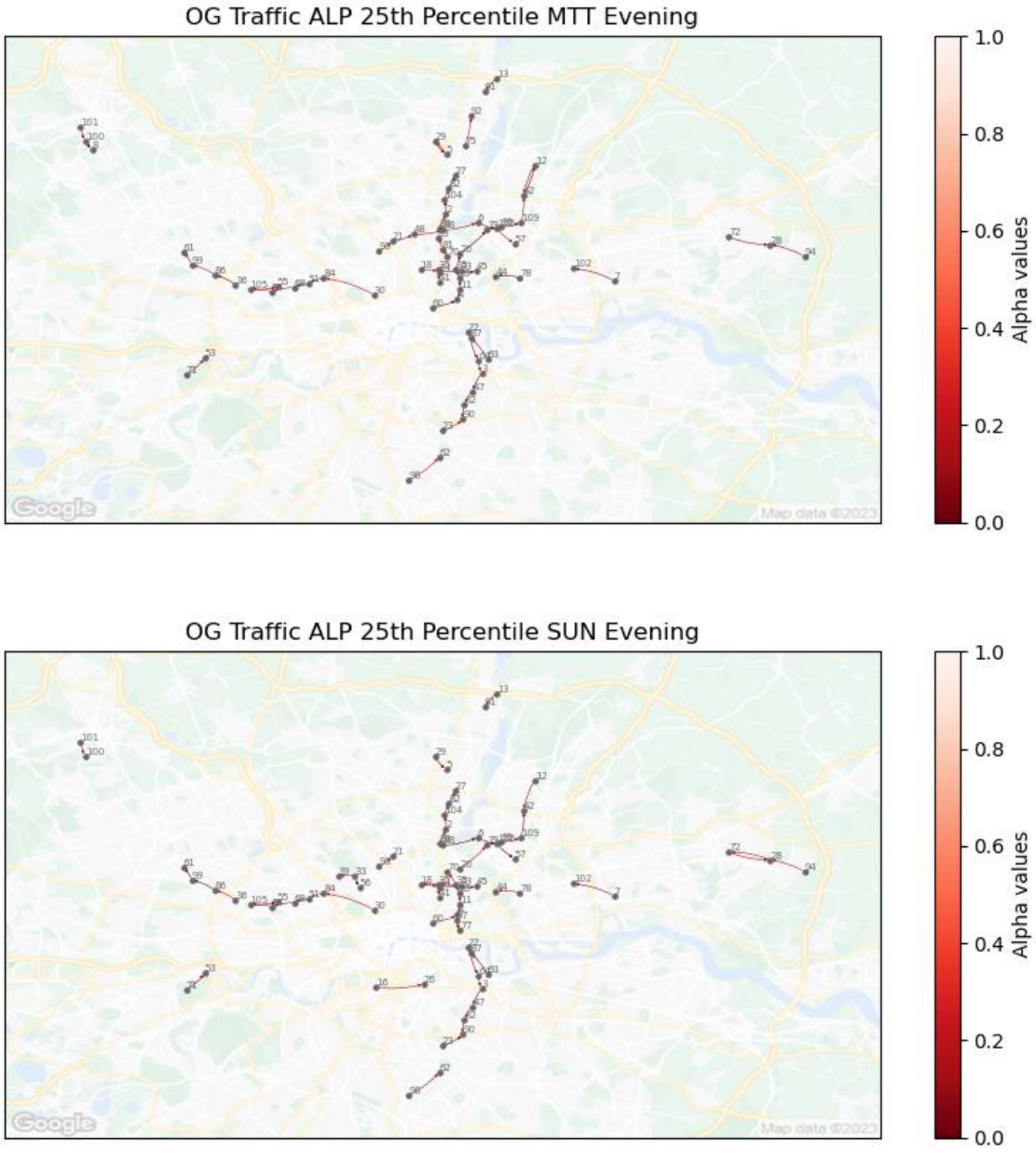

FIGURE A.18: Overground traffic 25th percentile disparity backbones.

A.6.2 Tube Traffic by Days

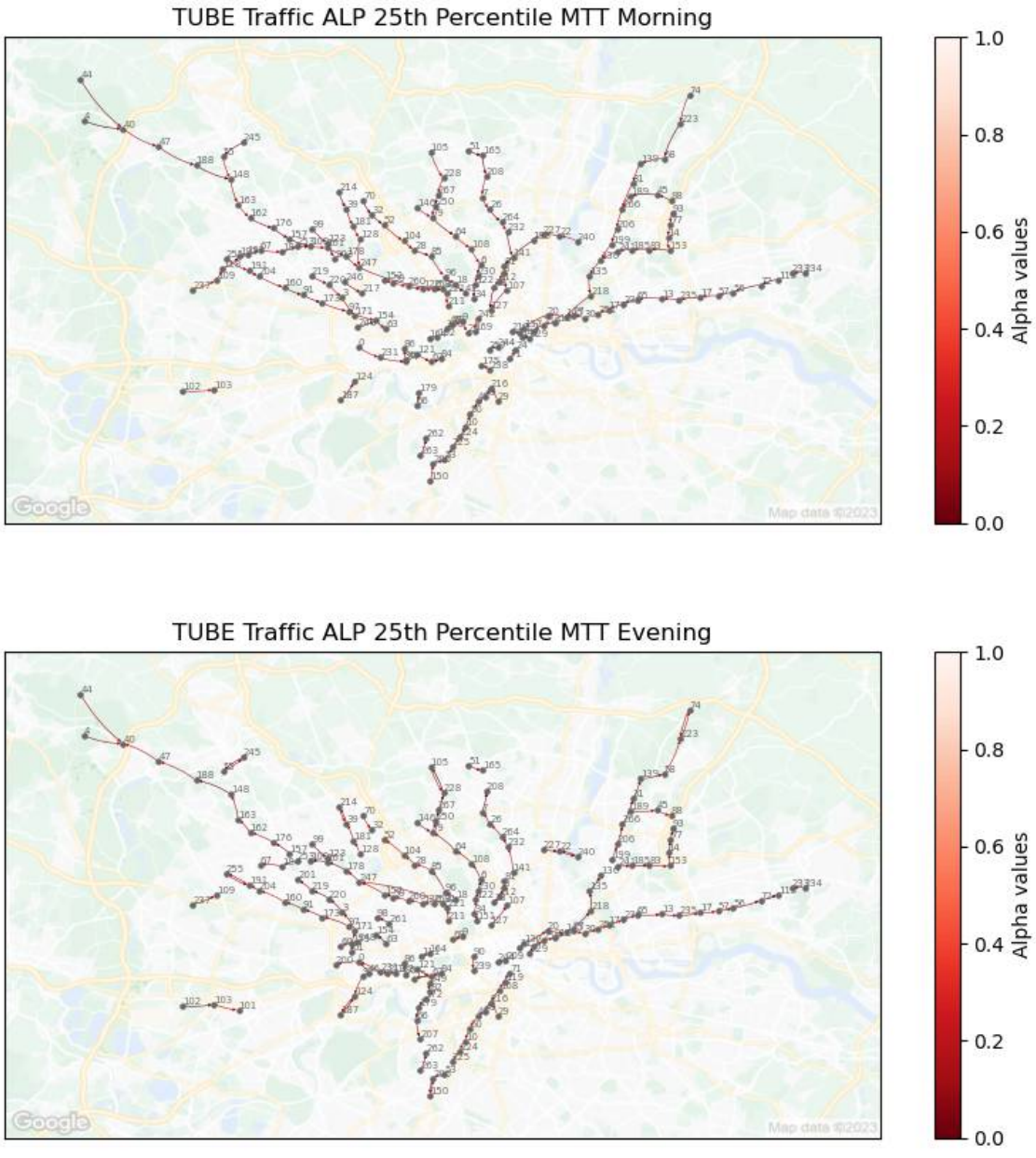

FIGURE A.19: Tube traffic 25th percentile thresholding backbones.

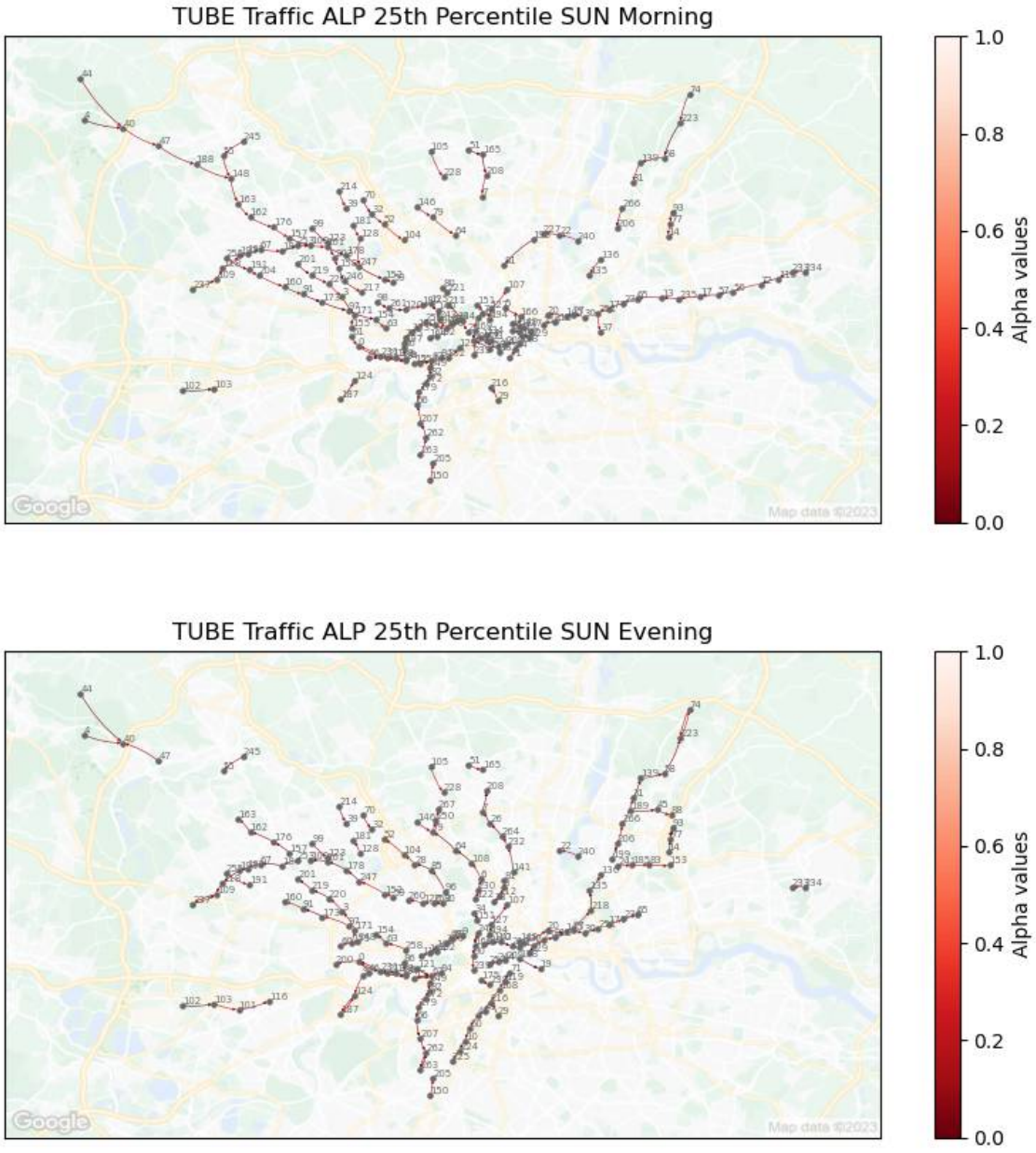

FIGURE A.20: Tube traffic 25th percentile disparity backbones.

A.7 Heatmap Traffic Correlation by Periods

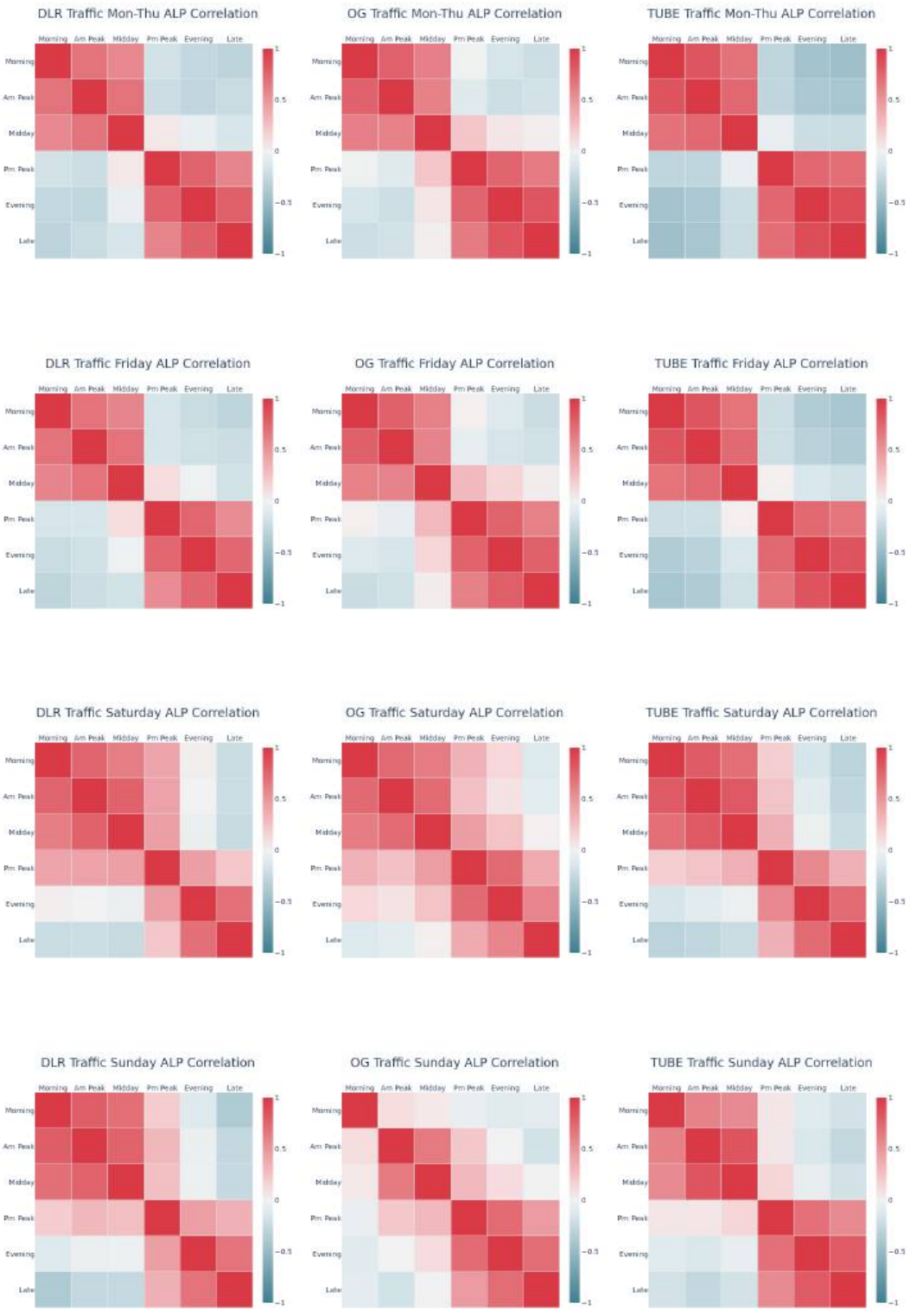

FIGURE A.21: Traffic ranks heatmap across the periods.

A.8 Heatmap Traffic Correlation by Days

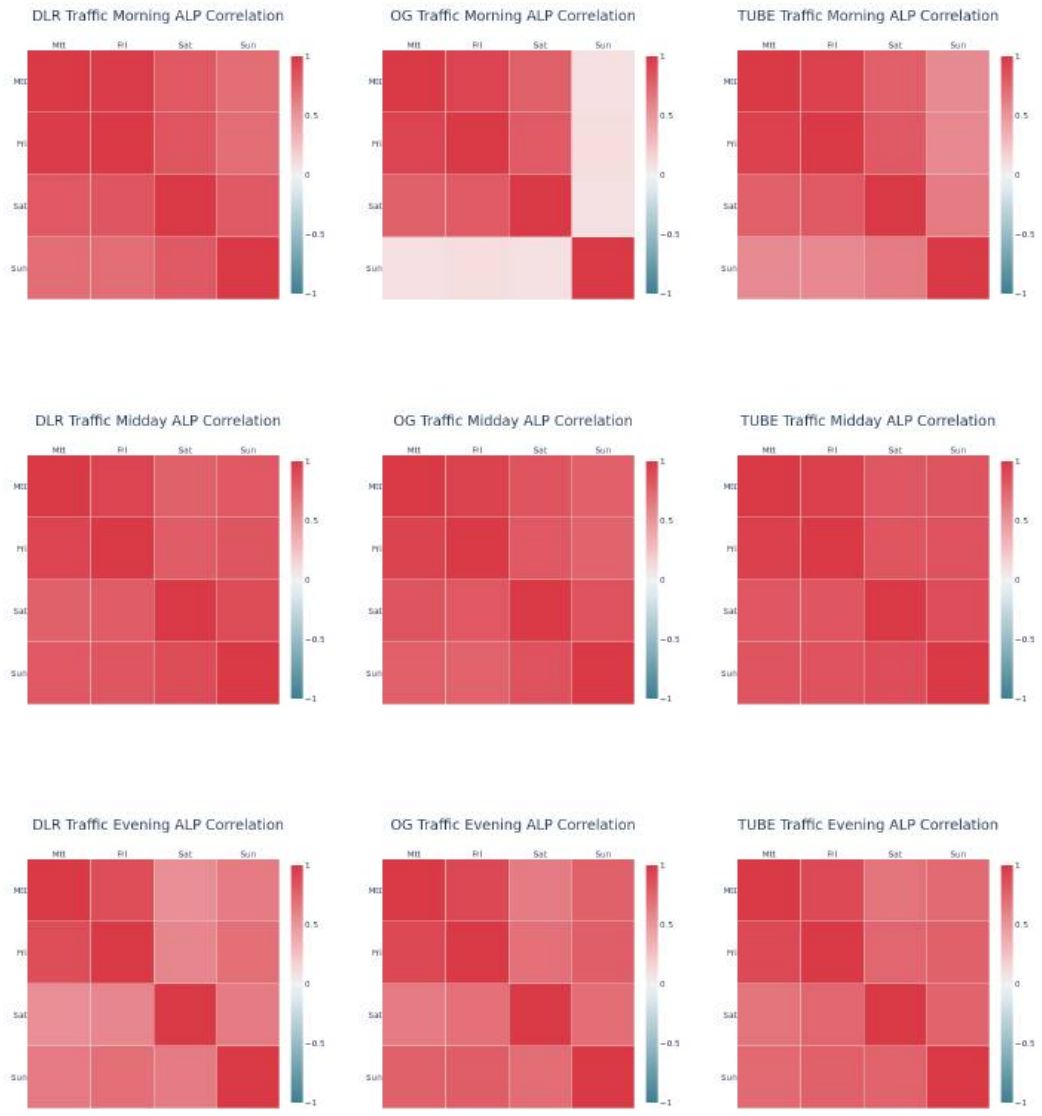

FIGURE A.22: Traffic ranks heatmap across the days.

A.9 Heatmap Efficiencies Correlation by Periods

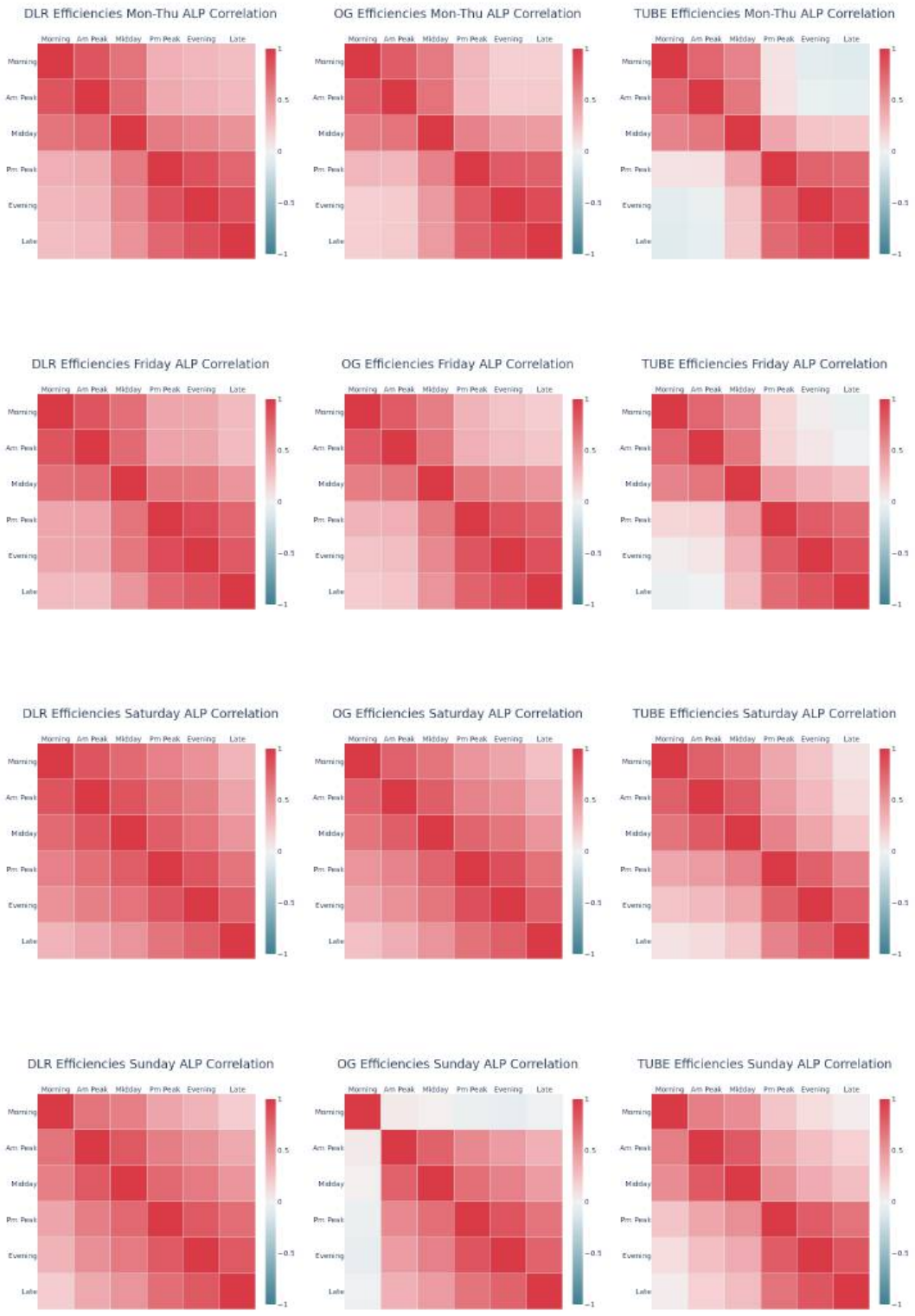

FIGURE A.23: Efficiencies ranks heatmap across the periods.

A.10 Heatmap Efficiencies Correlation by Days

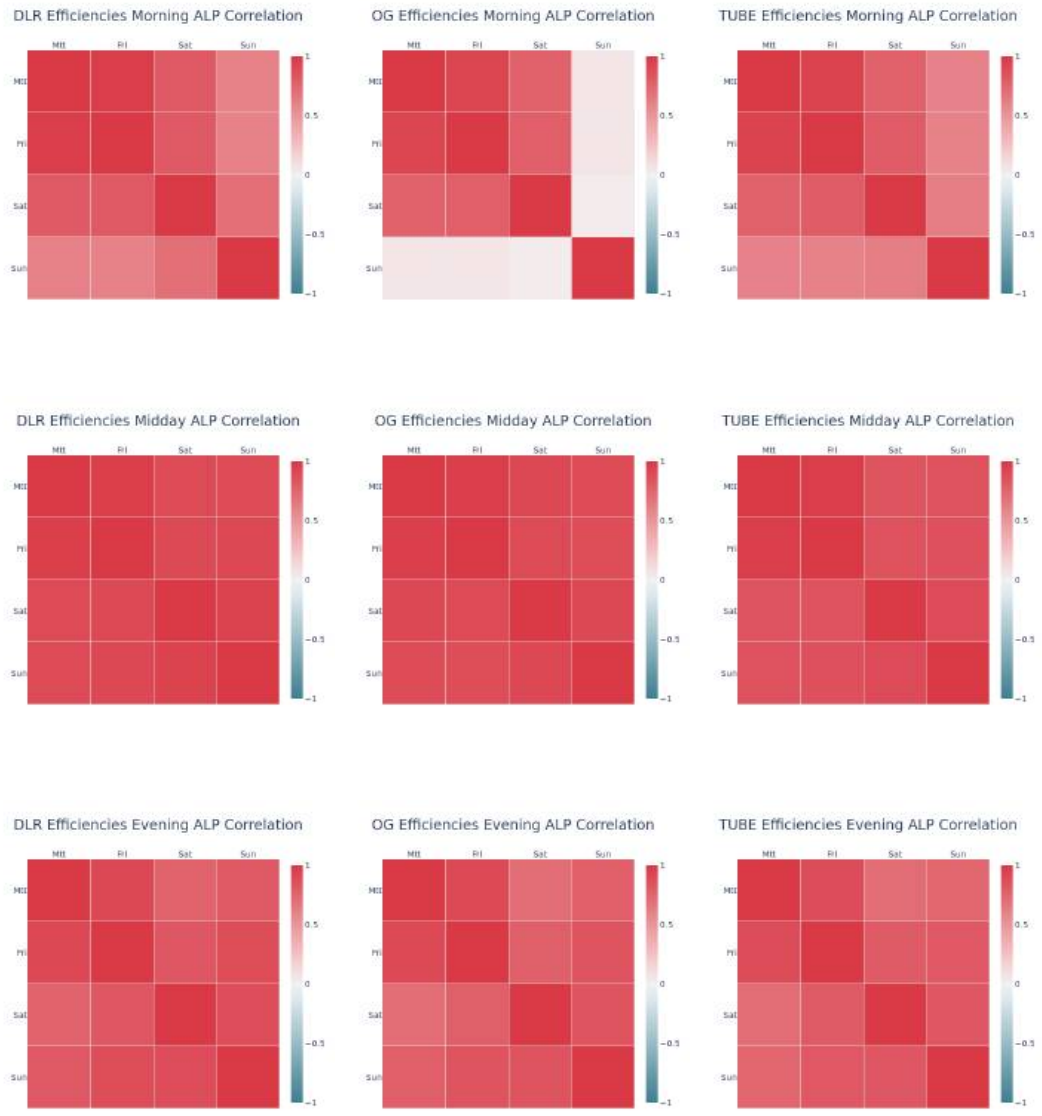

FIGURE A.24: Efficiencies ranks heatmap across the days.

A.11 Heatmap Distress Correlation by Periods

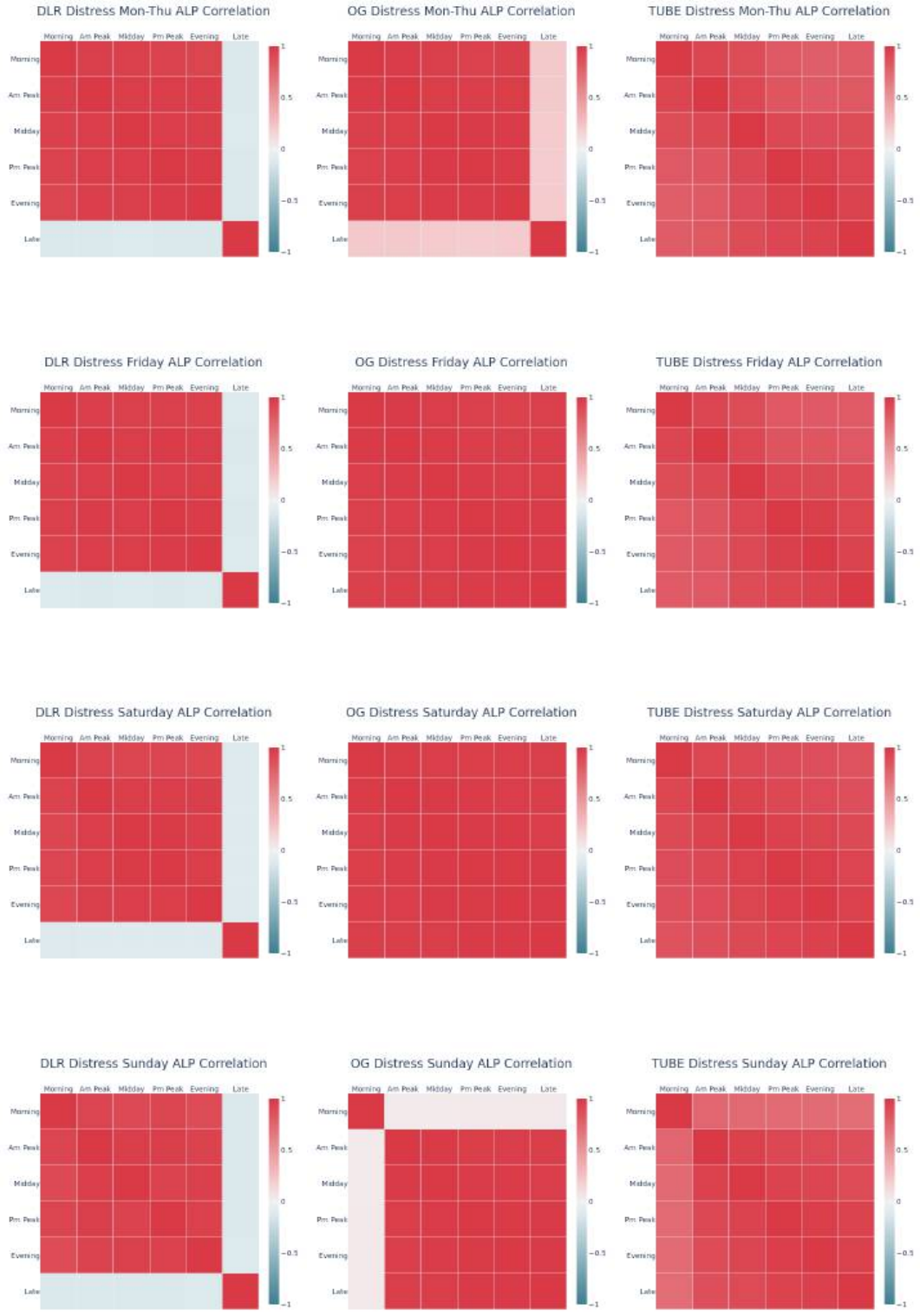

FIGURE A.25: Distress ranks heatmap across the periods.

A.12 Heatmap Distress Correlation by Days

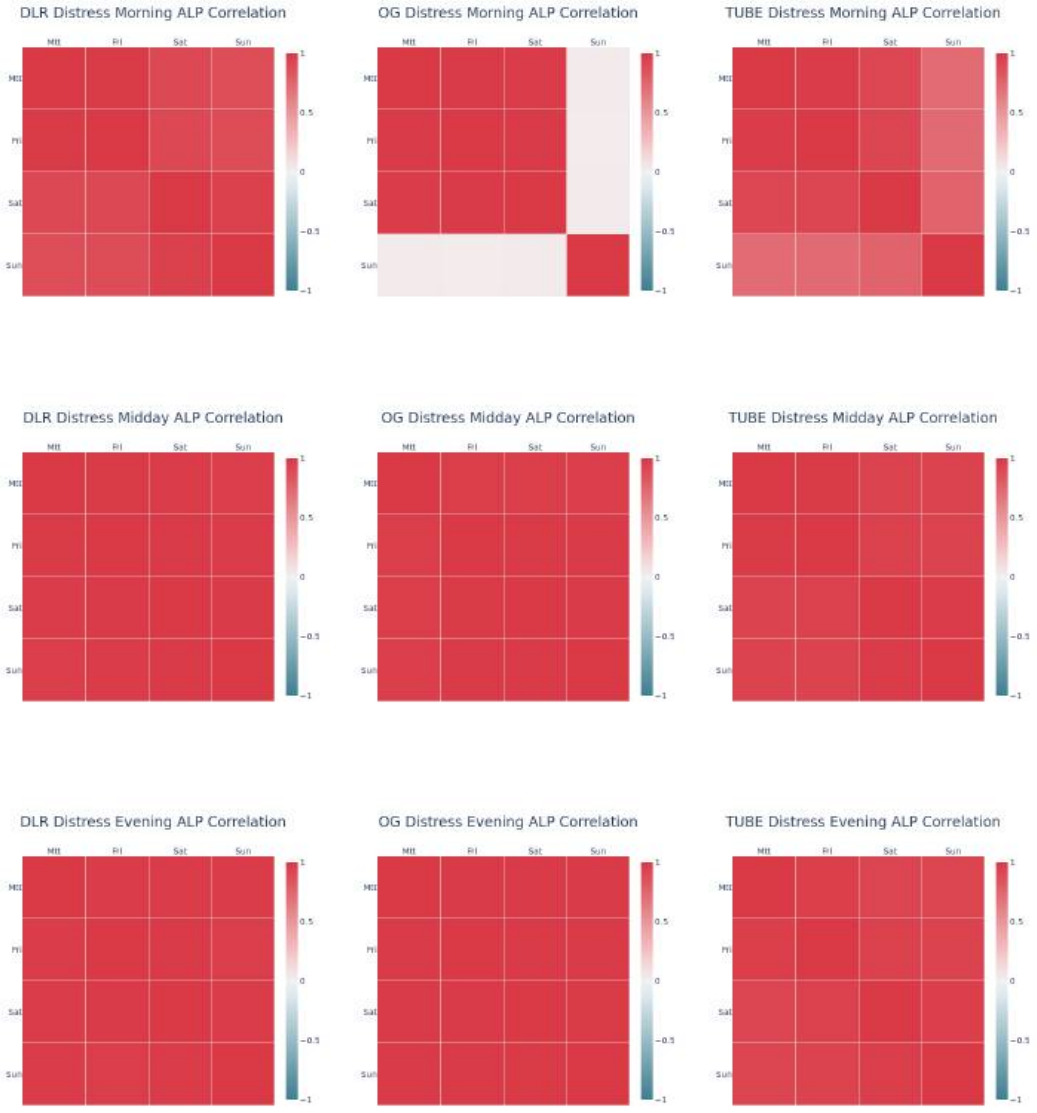

FIGURE A.26: Distress ranks heatmap across the days.

A.13 Efficiencies and Robustness Heatmap Correlation

FIGURE A.27: Efficiencies and robustness ranks correlation heatmap across the periods.

A.14 Efficiencies and Robustness Backbones

A.14.1 Overground Backbones

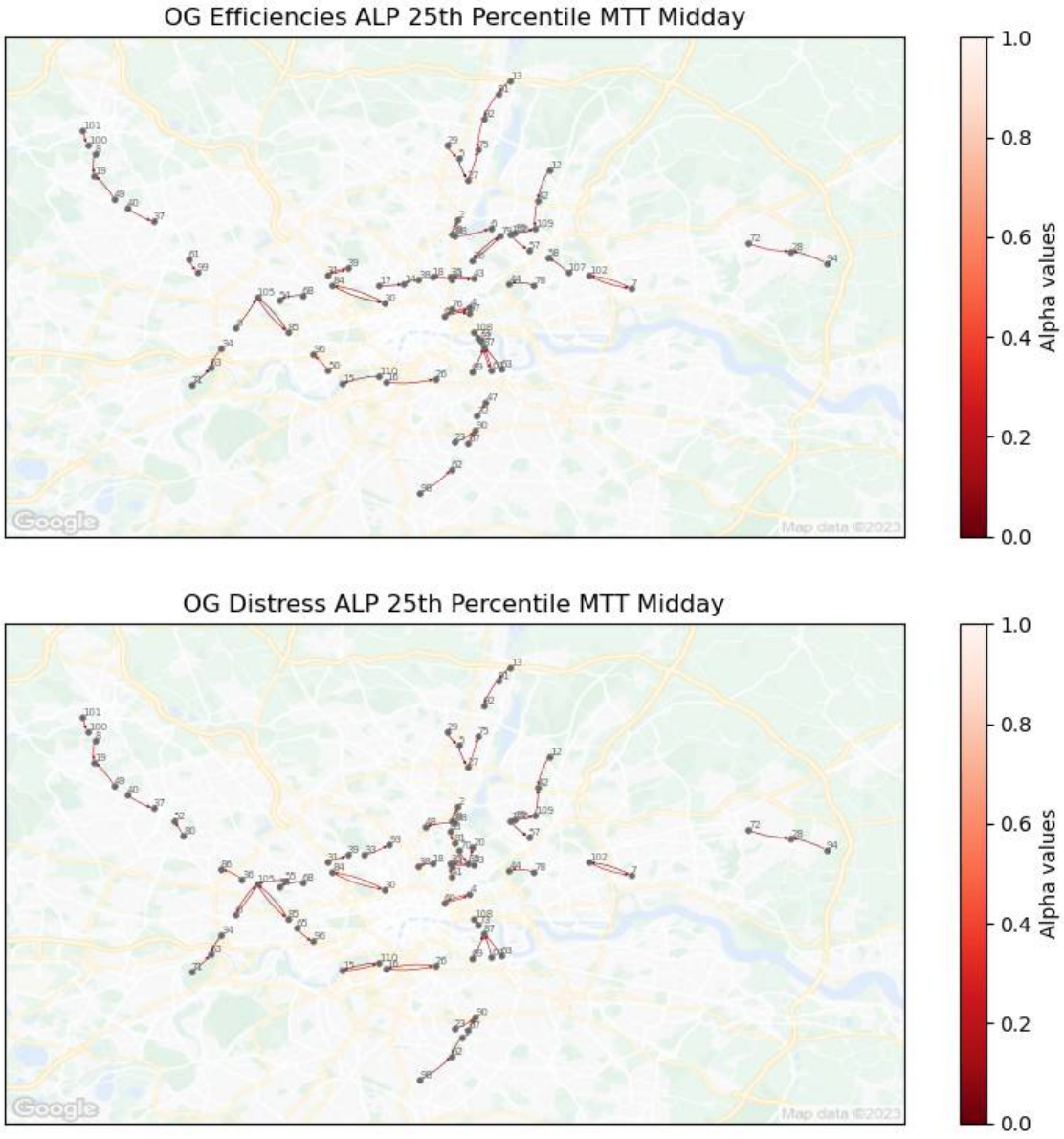

FIGURE A.28: Overground distance rankings.

A.14.2 Tube Backbones

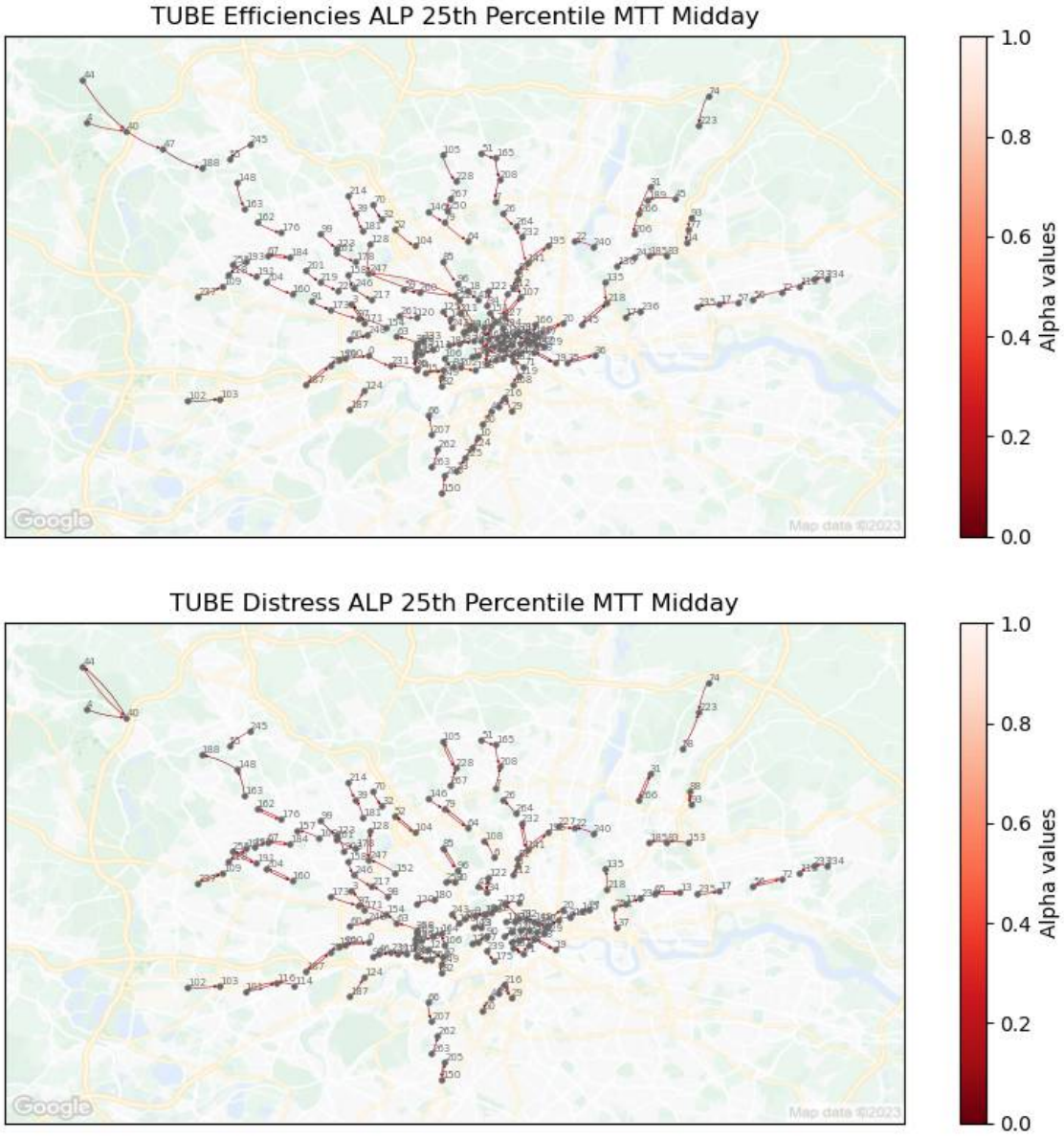

FIGURE A.29: Overground distance 25th percentile backbones.

Appendix B

B.1 Data Collection

```
from tfl.client import Client
from tfl.api_token import ApiToken
# User's Infiormations
app_id = '12345'
app_key = '12345'
path = r'C:\\Users\\Transport_Big_Cities\\'

### PROJECT'S ENDPOINTS ### (https://github.com/dhilmathy/TfL-python-api)
api_token = { app_id: token.app_id, app_key: token.app_key }
token = ApiToken(app_id, app_key)
client = Client(token)

### PROJECT'S ENDPOINTS ###
base_url = 'https://api.tfl.gov.uk/'
endpoints = {'stopPointByline': 'Line/{0}/Route/Sequence/all',
             ...,
             'time': '/Journey/JourneyResults/{0}/to/{1}?mode={2}&'}

### REQUEST VALIDATION ###
def get_query_strings(params):
    if params is None:
        params = {}
    if api_token is not None:
        params.update(api_token)
    return urlencode(params)
def send_request(location, params=None):
    return requests.get(base_url + location + '?' + get_query_strings(params))
```

```
### GET EXAMPLES ###
def get_stop_points_by_lineid(lineid):
    response = send_request(endpoints['stopPointByline'].format(lineid))
    data = response.json()
    return data
def get_stop_points_crowd_live(naptanid):
    response = send_request(endpoints['crowndingBynaptan'].format(naptanid))
    data = response.json()
    return data
def get_line_crowd_by_naptan(naptanid, lineid, direction):
    response = send_request(endpoints['crowndingByline'].format(naptanid,
    lineid, direction))
    data = response.json()
    return data

### EXTRACTION EXAMPLE ###
df_route = pd.DataFrame(columns=['lineId','modeName','name', 'direction','
    origination_name', 'destination_name', 'orig_naptan','dest_naptan'])

errors_route = []
for mode in modes:
        mode_routes = client.get_route_by_mode(mode)
        for mode_route in mode_routes:
            for route_section in mode_route.route_sections:
                try:
                    df_route = df_route.append({'lineId':mode_line.id,
                    'modeName':mode_line.mode_name,
                    'name':route_section.name,
                    'direction':route_section.direction,
                    'origination_name':route_section.origination_name,
                    'destination_name':route_section.destination_name,
                    'orig_naptan':route_section.originator,
                    'dest_naptan':route_section.destination},ignore_index =
    True)
                except:
                    errors_line.append((mode,mode_route, route_section))
```

Listing B.1: TFL API data collecting using Python language.

B.2 NEO4J

```
def create_db(db):
''' Connects to session in NEO4J graph database '''
    create_db_cypher = "CREATE DATABASE `"+db+"`"
    driver = GraphDatabase.driver("bolt://localhost:7687", auth=("neo4j",'x'))
    with driver.session() as session:
        result = session.run(create_db_cypher)
    session.close()
    driver.close()
    return
def generate_nodes(db, mode):
''' Generate nodes in the graph database '''
    path = "C:/buildbr/big-cities-transport/01.BaseGraph/02.Neo4J_Scripts/"
    create_nodes_cypher = open(path+mode+'_nodes.txt', 'r')
    with open(path+mode+'_nodes.txt', 'r') as file:
        create_nodes_cypher = file.read().rstrip()
    driver = GraphDatabase.driver("bolt://localhost:7687/", auth=('neo4j','x'))
    with driver.session(database=db) as session:
        result = session.run(create_nodes_cypher)
    session.close()
    driver.close()
    return
def generate_edges(db, kpi, dict):
''' Generate relation to pre-existing nodes in the graph database '''
    driver = GraphDatabase.driver("bolt://localhost:7687/", auth=('neo4j','x'))
    with driver.session(database=db) as session:
        id_pairs = list(dict.keys())
        for id_pair in id_pairs:
            ids= id_pair
            start, end = to_base(int(ids[0])+1,len(BS)), to_base(int(ids[1])+1,
    len(BS))
            kpi_value = dict[id_pair]
            seg = to_base(int(ids[0])+1,len(BS))+"_"+to_base(int(ids[1])+1,len(
    BS))
            create = "MATCH ({0}),({1}) WHERE ID({0})={2} AND ID({1})={3}
    CREATE ({0})-[{4}:TO{{{5}:{6}}}]->({1})".format(start, end, ids[0], ids[1],
     seg, kpi, kpi_value)
            result = session.run(create)
```

Listing B.2: Models Creation in NEO4J using Python language.

B.3 Disparity Filter Algorithm

```
def disparity_filter (graph, kpi):
    """ Disparity filter ALgo (https://arxiv.org/pdf/0904.2389.pdf)
        Inspiration: https://github.com/DerwenAI/disparity_filter/tree/main"""
    alpha_measures = []
    kpi_measures = []
    for node_id in graph.nodes():
        node = graph.nodes[node_id]
        degree = graph.degree(node_id)
        strength = 0.0
        for id0, id1 in graph.edges(nbunch=[node_id]):
            edge = graph[id0][id1]
            strength += edge[kpi]

        node["strength"] = strength
        for id0, id1 in graph.edges(nbunch=[node_id]):
            edge = graph[id0][id1]
            norm_weight = 0.0001 if edge[kpi] == 0 else edge[kpi] / strength
            #norm_weight = edge[kpi] / strength # divide
            aux = 'norm_'+kpi
            edge[aux] = norm_weight
            if degree > 1:
                try:
                    if norm_weight == 1.0:
                        norm_weight -= 0.0001
                    alpha = get_disparity_significance(norm_weight, degree)
                except AssertionError:
                    report_error("disparity {}".format(repr(node)), fatal=True)
                edge["alpha"] = alpha # adding alpha
                alpha_measures.append(alpha)
                kpi_measures.append(edge[kpi])
            else:
                edge["alpha"] = 0.0
    for id0, id1 in graph.edges():
        edge = graph[id0][id1]
        edge["alpha_ptile"] = percentileofscore(alpha_measures, edge["alpha"])
    / 100.0
        edge[f'{kpi}_ptile'] = percentileofscore(kpi_measures, edge[kpi]) /
    100.0
```

Listing B.3: Disparity Filtering in Python labguage.

B.4 Model Rankings

```
def period_rank_correlation_matrix(df):
    ''' Calculate Kendall ranking metric for peridos comparison '''
    for day in days:
        # Build matrixex
        kd_matrix_kpi = [[0] * len(periods) for _ in range(len(periods))]
        kd_matrix_alpha = [[0] * len(periods) for _ in range(len(periods))]
        kd_matrix_method = [[0] * len(periods) for _ in range(len(periods))]
        # Populate the matrix
        for i, period1 in enumerate(periods):
            for j, period2 in enumerate(periods):
                list1 = df.loc[:, (day, period1, f'{kpi}_rnk')].values #kpi1
                list2 = df.loc[:, (day, period2, f'{kpi}_rnk')].values #kpi2
                list3 = df.loc[:, (day, period1, f'alpha_rnk')].values #alpha1
                list4 = df.loc[:, (day, period2, f'alpha_rnk')].values #alpha2
                kd_matrix_kpi[i][j], p_value = kendalltau(list1,2)
                kd_matrix_alpha[i][j], p_value = kendalltau(list3,4)
                kd_matrix_method[i][j], p_value = kendalltau(list3,1)
        # Titles
        od_ = 'OG' if od == 'overground' else od
        title_method_kd = f'{od_.upper()} {kpi.title()} {days_[day]} ALP & THR
    Correlation'
        title_alpha_kd = f'{od_.upper()} {kpi.title()} {days_[day]} ALP
    Correlation'
        title_kd = f'{od_.upper()} {kpi.title()} {days_[day]} THR Correlation'
        # Plot the matrix
        heat_map(kd_matrix_kpi, od, title_kd, periods)
        heat_map(kd_matrix_alpha, od, title_alpha_kd, periods)
        heat_map(kd_matrix_method, od, title_method_kd, periods)
def day_rank_correlation_matrix(df):
    ''' Calculate Kendall ranking metric for days comparison '''
    for period in periods:
        kd_matrix = [[0] * len(days) for _ in range(len(days))]
        kd_matrix_alpha = [[0] * len(days) for _ in range(len(days))]
        for i, day1 in enumerate(days):
            for j, day2 in enumerate(days):
                list1 = df.loc[:, (day1, period, f'{kpi}_rnk')].values
                list2 = df.loc[:, (day2, period, f'{kpi}_rnk')].values
                list3 = df.loc[:, (day1, period, f'alpha_rnk')].values
                list4 = df.loc[:, (day2, period, f'alpha_rnk')].values
                # Matrix
                kd_matrix[i][j], p_value = kendalltau(list1, list2)
                kd_matrix_alpha[i][j], p_value = kendalltau(list3, list4)
```

```
        # Titles
        od_ = 'OG' if od == 'overground' else od
        title_alpha_kd =f'{od_.upper()} {kpi.title()} {period} ALP Correlation'
        title_kd = f'{od_.upper()} {kpi.title()} {period} THR Correlation'
        # Plot the matrix
        heat_map(kd_matrix, od, title_kd, days)
        heat_map(kd_matrix_alpha, od, title_alpha_kd, days)
def kpis_rank_correlation_matrix(df1, df2):
    ''' Calculate Kendall ranking metric for kpis comparison'''
    for day in days:
        kd_matrix_kpis = [[0] * len(periods) for _ in range(len(periods))]
        for i, period1 in enumerate(periods):
            for j, period2 in enumerate(periods):
                list3 = df1.loc[:, (day, period1, f'alpha_rnk')].values
                list4 = df2.loc[:, (day, period2, f'alpha_rnk')].values
                kd_matrix_kpis[i][j], p_value = kendalltau(list3, list4)
        # Titles
        od_ = 'OG' if od == 'overground' else od
        title_kpis_kd = f'{od_.upper()} ALP Eff & Robust {days_[day]}
    Correlation'
        # Plot the matrix
        heat_map(kd_matrix_kpis, od, title_kpis_kd, periods)
```

Listing B.4: Kendall Correlation Comparison in Python language.

Bibliography

[1] Merlin Stone and Eleni Aravopoulou. Improving journeys by opening data: the case of transport for london (tfl). page 2–15, January 2018. URL https://doi.org/10.1108/BL-12-2017-0035.

[2] Marián Boguñáb M. Ángeles Serranoa, 1 and Alessandro Vespignanic. Extracting the multiscale backbone of complex weighted networks. *PNAS*, 106(16), April 2009. URL www.pnas.org/cgi/doi/10.1073/pnas.0808904106.

[3] Eric Guichard Marc Barthelemy, Bernard Gondran. Spatial structure of the internet traffic. *Physica A*, 319(319):633–642, March 2003. URL https://doi.org/10.1016/S0378-4371(02)01382-1.

[4] Stephen Marshalla Ed Manley Yuerong Zhanga, b. Network criticality and the node-place-design model: Classifying metro station areas in greater london. *Journal of Transport Geography*, (79), July 2019. URL https://doi.org/10.1016/j.jtrangeo.2019.102485.

[5] HervéAbdi. The kendall rank correlation coefficient. January 2006. URL https://personal.utdallas.edu/~herve/Abdi-KendallCorrelation2007-pretty.pdf.

[6] José J. Ramasco Filippo Radicchi and Santo Fortunato. Information filtering in complex weighted networks. *Physical Review*, January 2011. URL 10.1103/PhysRevE.83.046101.

[7] Han Zhao1 Qi Wang1 Ji Zhu Xiaohang Zhang1*, Zecong Zhang1. Extracting the globally and locally adaptive backbone of complex networks. *PLoS ONE*, June 2014. URL https://doi.org/10.1371/journal.pone.0100428.

UNIVERSITÉ CATHOLIQUE DE LOUVAIN
École polytechnique de Louvain
Rue Archimède, 1 bte L6.11.01, 1348 Louvain-la-Neuve, Belgique | **www.uclouvain.be/epl**