Exploring Subway Relational Model in Decision Support Systems

Sondoss Chtioui, Sébastien Saudrais, Sebti Mouelhi, Toufik Azib, Marc Ille, Mélanie Morel, Alexandre Rossi, Jerome Charmetant

Emerging Cutting-Edge Developments in Intelligent Traffic and Transportation Systems
M. Shafik (Ed.)

doi:10.3233/ATDE240036

Exploring Subway Relational Model in Decision Support Systems

Sondoss CHTIOUI [a,b,1] and SébastienSAUDRAIS [a]
Sebti MOUELHI [a] , Toufik AZIB [a]
Marc ILLE [b] , Melanie MOREL [b]Alexandre ROSSI [b], Jerome CHARMETANT [b]
[a] *ESTACA, ESTACA'Lab, LavaL & Paris-Saclay,* France
[b] *EGIS Rail, Lyon, France*

Abstract. This paper examines recent research in the field of public transportation, specifically focusing on the development of learning algorithms for predicting the behavior of trains and buses. However, it underscores the overlooked significance of having a clear and structured representation of data entities. To address this oversight, a relational model is proposed that captures the essential data fields specific to the subway system to enhance the learning process. The model undergoes validation through collaboration with a metro control center and domain experts. Furthermore, the study integrates this relational model into a hybrid approach that combines online and offline machine learning techniques. This approach effectively forecasts delays and passenger flow, thereby enabling informed decision-making and optimizing rail operations through a decision support system. The paper concludes by emphasizing the pivotal role of the proposed model in facilitating the selection of relevant variables for each learning problem.

Keywords. Railway Operation, Subway Lines, Control Center, Planned Timetable (PTT), Real Timetable (RTT), Data Modeling, UML, Class Diagram, Machine Learning (ML), Passenger load, Delay.

1. Introduction

Train regulation plays a pivotal role in ensuring the efficiency and safety of public transportation systems. By managing train frequency and speed, it minimizes waiting times for passengers and optimizes resource utilization. Controlling train speeds also reduces the risk of accidents and collisions, enhancing overall safety. Moreover, these regulations are instrumental in improving passenger flow management at various stations, preventing overcrowding, and minimizing delays. Railway control systems, consisting of sophisticated hardware and software applications, monitor train positions and movements in real-time to ensure network safety. These systems facilitate the exchange of diverse data formats, including audio, video, scheduling mode, and sensor data like speed, location, and track occupancy. Each train is assigned a unique number corresponding to a pre-programmed movement line within specific zones, monitored by dedicated control center operators. In case of unforeseen events, such as delays or disruptions, operators promptly devise strategies to manage affected rail lines and alleviate congestion, working in coordination with the respective rail control centers. This seamless integration of regulations and monitoring systems ensures effective train regulation and smooth railway operations for a reliable and passenger-friendly transportation experience.

[1] Sondoss CHITIOUI, ESTACA, ESTACA'Lab , Campus Ouest, Rue Georges Charpak - 53000 Laval, France; E-mail: Sondoss.chtioui@estaca.fr.

Integration of machine learning in train regulation for subway line is an important technological advancement that can provide numerous benefits. By using machine learning algorithms, it is possible to predict passenger wait times based on various factors such as time of day, day of the week, weather conditions, and more. This prediction allows for the real-time regulation of train frequency and speed to minimize passenger wait times and improve the efficiency of the public transportation system [1]. Machine learning can also help avoid overcrowding and delays by adjusting train frequency and speed based on live data such as passenger demand and traffic conditions and improve the efficiency, safety, and reliability of the public transportation system [2]. This can allow passengers to travel faster, easier, and more comfortably while reducing the costs and environmental impacts associated with public transportation systems. Indeed, most of the existing works on ML for subway systems are purely theoretical and have not yet been tested in real-world conditions. Researchers tend to focus on mathematical modeling of the system, which can be useful for predicting theoretical performance and outcomes. However, these works often do not consider the many practical aspects that can influence system efficiency and reliability. For instance, the quality of the data and the selection of variables are of paramount importance. Noisy, incomplete, or incorrect data can lead to unreliable results. Additionally, a judicious selection of features can enhance the efficiency and reliability of the model by eliminating redundant or uninformative characteristics. On the other hand, the choice of algorithms and their parameters is crucial. Furthermore, a significant aspect in our work is reinforcing the prediction model with real-time mathematical calculations to validate the predictions at time *t*.

Railway data learning and modeling are constantly evolving fields that have practical applications in rail traffic management, predictive infrastructure maintenance, passenger safety [3], [4]. Numerous studies have been conducted in the literature to improve railway systems performance and efficiency by exploiting data generated by various equipment such as sensors, signaling systems [5],[6]. Machine learning techniques, especially neural networks, are widely used to model and predict railway outputs (delay, passenger,). Linear models have been mostly superseded by complex models [7], [8], including deep neural networks to predict train arrival delay using extreme learning machine(ELM) with nine characteristics plus the particle swarm optimization (PSO) algorithm to optimize the hyperparameter of the ELM, that have greater accuracy and performance, and have the ability to extract valuable insights from unprocessed and unstructured data using gradient boosting (XGBoost) prediction model that captures the relation between the train arrival delays and various railway system characteristics. In conjunction with the notable progress of artificial intelligence, the real-time analysis of passenger data continues to represent a rapidly expanding area of research that cannot be overlooked [9]. The latest technological advances allow large volumes of data to be processed and analyzed in real time, enabling operators to make knowledgeable rapid decisions enabling a more efficient rail traffic and enhanced passenger safety [10]. To summarize, the current state of rail data learning and modeling involves the increased utilization of machine learning techniques, integration of multimodal techniques using three different methods to define inputs including normalized real number, binary coding, and binary set encoding inputs [11] and real-time data analysis. These advances have significant implications for optimizing rail system management and enhancing passenger safety. Using learning algorithms to predict delays and passenger flow is an essential component of a decision support system that assists center operators in making optimal decisions regarding time and resources.

In this paper, the objective is to integrate machine learning techniques into the management of subway traffic control, with the potential to improve system efficiency and safety while reducing wait times and delays. It is crucial to collaborate closely with domain experts to successfully implement machine learning techniques into metro traffic control management. This will enable us to gain a deeper understanding of the unique challenges and opportunities of this system and develop practical and reliable solutions that can be effectively applied in the real world. The proposed approach marks the first stage of our work and serves as a crucial step in constructing a dependable machine learning system for managing railway traffic. It guarantees the availability of precise and comprehensive data to the system, which is essential for precise predictions and informed decision-making. Before implementing algorithms, the first step is to define the system's characteristics. By creating a relational model, we can identify all the elements and their interconnections. This helps us understand which elements are necessary for accurate predictions of delays and passenger flow. We can explicitly comprehend the relationships and correlations between various variables. This understanding is important for the development of efficient machine learning models enabling accurate forecast of future events [12].

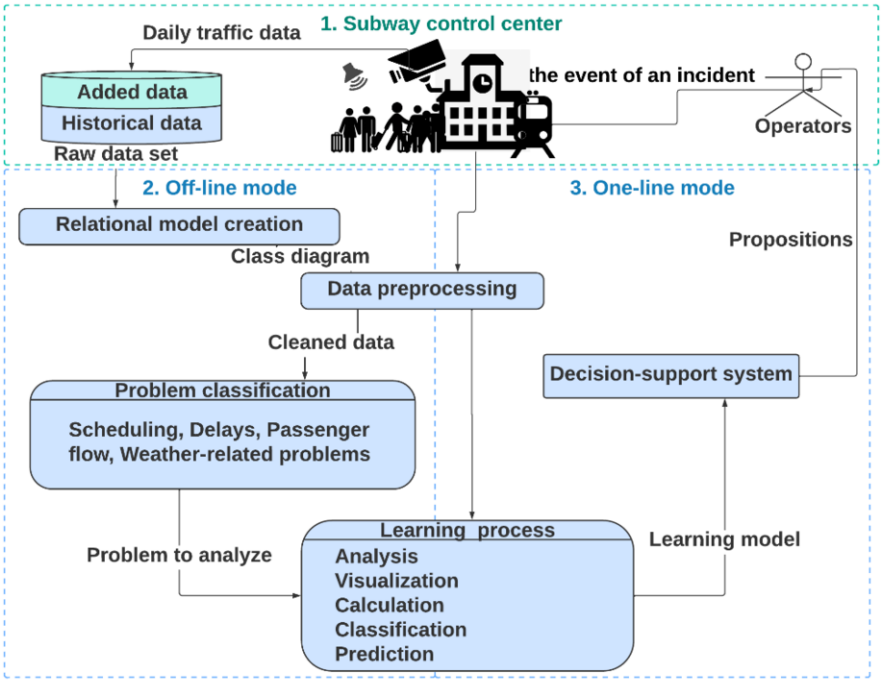

Figure 1. Diagram illustrating the methodology.

The paper manuscript is organized as follows. Section “Methodology” outlines our approach undertaken to achieve the modeling objective. It starts with a comprehension of the nature of raw data provided by the control centers and progresses towards identifying the useful data for machine learning. Section “Relational model creation” presents our proposed relational model represented with à class diagram. Section “Case study”

presents the application of the relational model to a use case of raw data from a control center. Section “Conclusion” summarizes the potential for future work that can be done using the modeled data approach to predict passenger flows based on delays.

2. Methodology

In this section, we present the hybrid approach of our work aimed at developing a decision support system for rail operators. The objective of this system is to help operators make quick and efficient decisions to manage a smoother rail traffic. The decision support system is the final step of our methodology, which is based on machine learning steps to predict delays and passenger flows, and then integrate them into the system. " Global methodology" section presents a figure that describes all the steps needed to achieve this hybrid approach. The next section, "Working method" focuses on the first step of the global methodology, which is the mean of our paper to create a relational model to bring together all the metropolitan data.

2.1. Global methodology

The methodology shown in Figure 1 presents a hybrid approach to machine learning based on a combination of offline and online mode learning. The article proposes the integration of this approach into the automatic learning algorithms used for subway line operations to predict the passenger flow and delay. The hybrid approach combines the strengths of both offline and online learning modes to achieve optimal performance in training machine learning models. This approach is applicable in various domains, including weather forecasting using NWP numerical weather prediction and observation in real time to correct the predictions [13].

In the railway control center, two distinct learning modes, offline and online, are employed to enhance the efficiency of the transportation system. The offline learning mode processes historical and real-time data from a comprehensive railway database to predict delays and passenger flow on platforms. Historical data aids in evaluating past system performance and identifying areas for improvement, while real-time data ensures the continuous monitoring and improvement of predictions. The online mode, on the other hand, verifies the reliability of offline predictions by comparing delays and intervals at a specific time with past data. This decision support system proposes suitable solutions to operators based on predictions of delays and passenger flows, taking into account factors such as current traffic conditions, system failures, and weather conditions, aiming to optimize operations and reduce the need for unnecessary delay regulations when platforms are empty.

Here is an example of general traffic regulation scenarios that could be taken into consideration in the system: during peak hours, there is a significant increase in passenger flow, and delays are anticipated. In this case, our decision support system suggests implementing traffic regulation measures to manage the increased demand. For instance, the system proposes adjusting train frequencies, optimizing timetables, or even adding extra trains to accommodate the higher number of passengers efficiently. It is important to note that the decision support system offers easy-to-understand solutions to operators. that could be taken into consideration in our system. The system considers the relationship between passenger flows and delays in order to propose appropriate solutions.

In this paper, we focus on the first phase of the hybrid approach presented in Figure1,

which is "relational model creation". It enables the creation of an accurate and comprehensive representation of the entire railway system, including equipment, infrastructure, vehicles, and operational processes. Understanding the nature of the data is a crucial step in modeling it effectively. Without a clear understanding of the data, it is difficult to select the appropriate data and algorithms for each classification problem.

2.2. Working method

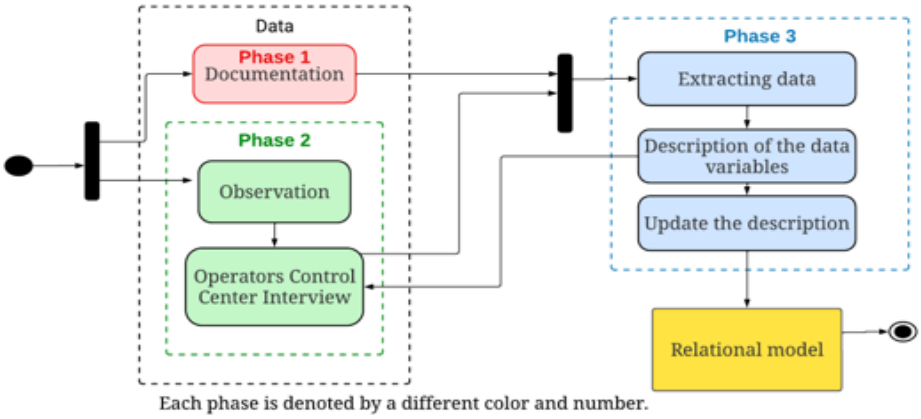

Figure 2. Schematic diagram demonstrating the process of data modeling.

Figure 2 illustrates the three steps of our working method that were employed to establish a relational model. The first phase is "Documentation» analysis, which involved analyzing similar articles to identify the inputs used in their algorithms to describe the data related to subway system constraints. The state of the art refers to the current state of knowledge, research, and practices in a particular field or area of study. The analysis of the state of the art can aid in simplifying complex datasets, making them more suitable for analysis. By dividing the dataset into subsets, researchers can uncover patterns and relationships that may not be immediately observable when considering the data.

Phase 2, based on discussions with control center operators, can provides valuable feedback on the relational models and their practical. Discussion with rail control center operators allows the model to be validated on real-world examples with people who are very familiar with the domain. After several discussions with the operators, we were able to extract more information than is possible with people who do not know the system. The better understanding of the data and its context made the modeling phase richer, but the validation with a control center will make the resulting models more easily applicable in real metro scenarios, as we seek to integrate a decision support system in such centers.

Phase 3 represents the recursive process of comprehending our data and defining a relational diagram as an output in the yellow rectangle (Figure 2) to facilitate the understanding of our data, as explained in the previous section.

The relational modeling of metro railway data involves organizing data related to a metro railway system into a set of tables, where each table represents a specific aspect of the system. The tables are then linked together through common data elements to establish relationships between them. For example, one table may contain data about the stations in the system, such as their names and locations, while another table may contain

data about the trains that run on the system, such as their schedules and capacities. By linking these tables through a common data element such as a station ID, it becomes possible to answer questions such as which trains are scheduled to stop at a particular station at a given time.

The next section describes the relational model used to link the data into an explanatory schema to facilitate data analysis.

3. Relational model creation

A class diagram is a static structure diagram in UML that is commonly used to represent object-oriented software systems. By combining the elements of class diagram, we can provide a comprehensive representation of the system's structure and behavior, making them useful for modeling and designing software systems [14]. By modeling the data, the first step is to identify the key concepts in the system and representing them as classes. Each class has a name, and it may also have attributes (data) and interface defines a set of methods that a class must implement. These components were used to define the structure and behavior of the system being modeled. In our case, we create a model for a subway system that includes several classes based on the different phases presented in Figure 2. Each class has properties described by attributes such as station ID, train ID, real and planned departure times.

In this section, we introduce a relational model of class diagram to present the variables and their relationships in an easily readable format. Building upon the phases depicted in Figure 2, we start with the section “phase 1: Documentation” which provides an initial representation of a basic class diagram. Then, in the section “Phase 2: Visits to the control center”, we present a more comprehensive class diagram that has been modified with the experts of the center. Finally, in section “Phase 3: Relational model”, we present the final representation of the class diagram, applicable to all the steps described in our hybrid approach presented in Figure 2.

3.1. Phase 1: Documentation

In the documentation, we have constructed the initial part of our class diagram that encompasses a set of data related to railway traffic. Phase 1 of the working method, presented in Figure 2, describes the work carried out during the state-of-the-art process to create a first-class diagram that describes the data we have successfully extracted based on existing articles on topics similar to our railway traffic subject, as well as internal documents from Egis that describe the structure of a railway line (subways, trams).The existing documents consist of articles on machine learning for predicting delays, incidents, and the relationships between delays and various characteristics of the railway system (such as scheduled arrival time at the station, influence of infrastructure defined by expert opinions, percentage of distance covered, actual distance traveled, travel time, and track). They also address challenges and open research problems in railway traffic management. We have used the input data explained in the state-of-the-art articles presented in the "Introduction" section to describe an initial set of classes presented in Figure 3, such that each article provides us with a set of variables and their relationships [7], [8], [15]. For example, a station should be linked with a line, track, and train, and the RTT should have an association of membership with them, ensuring that we cannot have an RTT without first having Line, Track, Station, and Train. Furthermore, the classes

Weight, Ticketing, and Camera are based on internal documents from Egis Rail, with the same association of membership between them and the station and train.

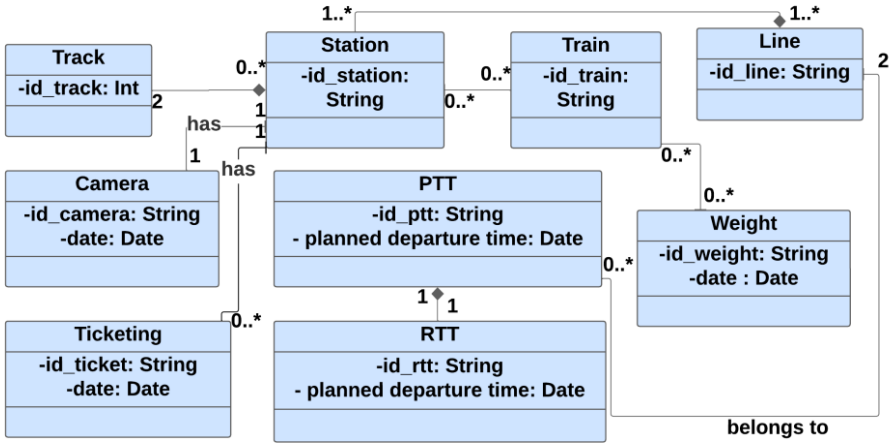

Figure 3. Initial class diagram of railway traffic.

3.2. Phase 2: Visits to the control center

Phase 2 of Figure 2 involves observations in railway control centers and interviews with operators, who are experts in the railway domain. This practical approach offers several advantages, including a better understanding of the system, real-time data collection, interaction with experts, and model validation. By directly observing the operations of the railway control centers, we were able to verify, compare, and refine our initial class diagram in Figure 3 by aligning it with the actual processes and operations of the control centers. This led to Figure 4, which presents an improved class diagram that has been validated by metro control center experts. In this revised class diagram, we have introduced additional calculation classes, such as Interstation, Information, and Interval, along with additional methods for the existing classes. These additions allow us to gather supplementary relevant information, such as intervals between consecutive trains and delays at previous stations. Furthermore, we have included the Day class to account for the influence of different types of days on the system, including events, weather conditions, and the day of the week. Control center experts have emphasized that these variables are crucial as they can impact delays and affect passenger congestion on station platforms.

By directly observing control centers and conducting interviews with expert operators, we have enhanced our modeling of metro data by incorporating more precise elements and validating our approach with industry practitioners.

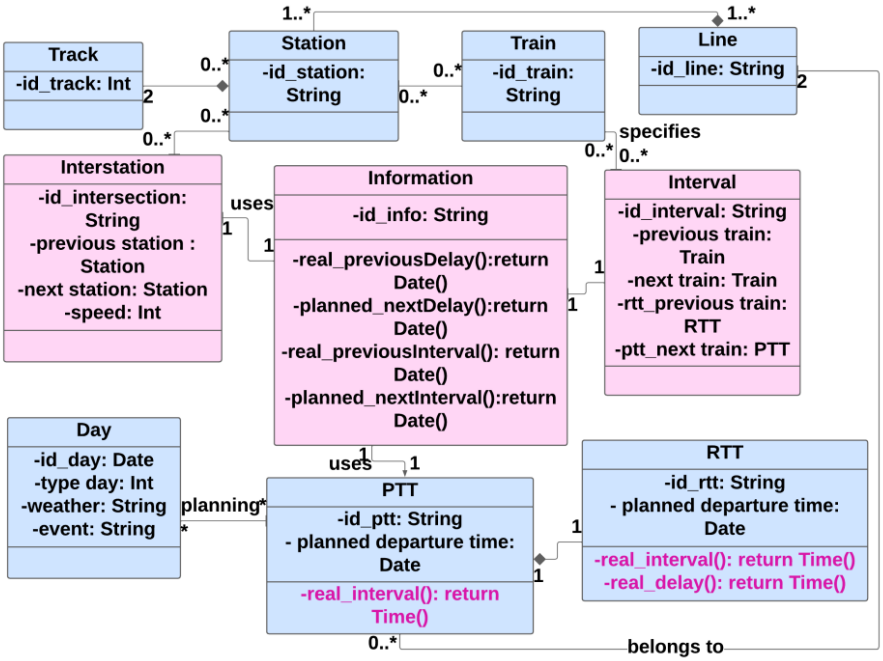

Figure 4. Refined and Validated Class Diagram by Metro Control Center Experts.

3.3. Phase 3: Relational model

Phase 3 of Figure 2 entails reformulating the class diagram to enhance its representation and separate the Operation and Passenger Flow subsystems. This reformulation serves as the basis for the subsequent phases of the hybrid approach presented in Figure 1, allowing the development of a decision support system. The utilization of this class model enables the determination of variable dependencies for addressing specific problems related to delays and passenger flows. By separating the classes into two packages Operation and Passenger Flow subsystems, the reformulated class diagram provides a clearer representation of the classes and their relationships within each subsystem. This diagram serves as a crucial foundation for the subsequent phases of the hybrid approach, enabling the development of a decision support system for various aspects of the railway system.

Utilizing this class model in the approach aids in selecting the variable dependencies for each specific problem, such as delays and passenger flows. By having a well-structured class model, it becomes easier to identify the relationships and interactions among different variables. This approach facilitates making informed decisions by leveraging the information contained within the class model.

The class diagram presented in Figure 5 illustrates all the classes and interfaces that make up the railway system of the subway line, which is divided into two separate packages - one for the "operating system" and the other for "passenger flow" This division aims to simplify the calculation of variables for each system separately, thereby enhancing the accuracy of the system. The "operating system" class is responsible for gathering all data related to traffic, such as Line, Station, Train, Track, Day, PTT, and RTT, which

are the raw data of the control center dataset. The Interval, Interstation, and Information classes are dedicated to calculating missing data in the raw dataset, such as delays, intervals, and previous trips, and their usefulness was validated during visits with operators of subway control centers, as explained in Figure 2.

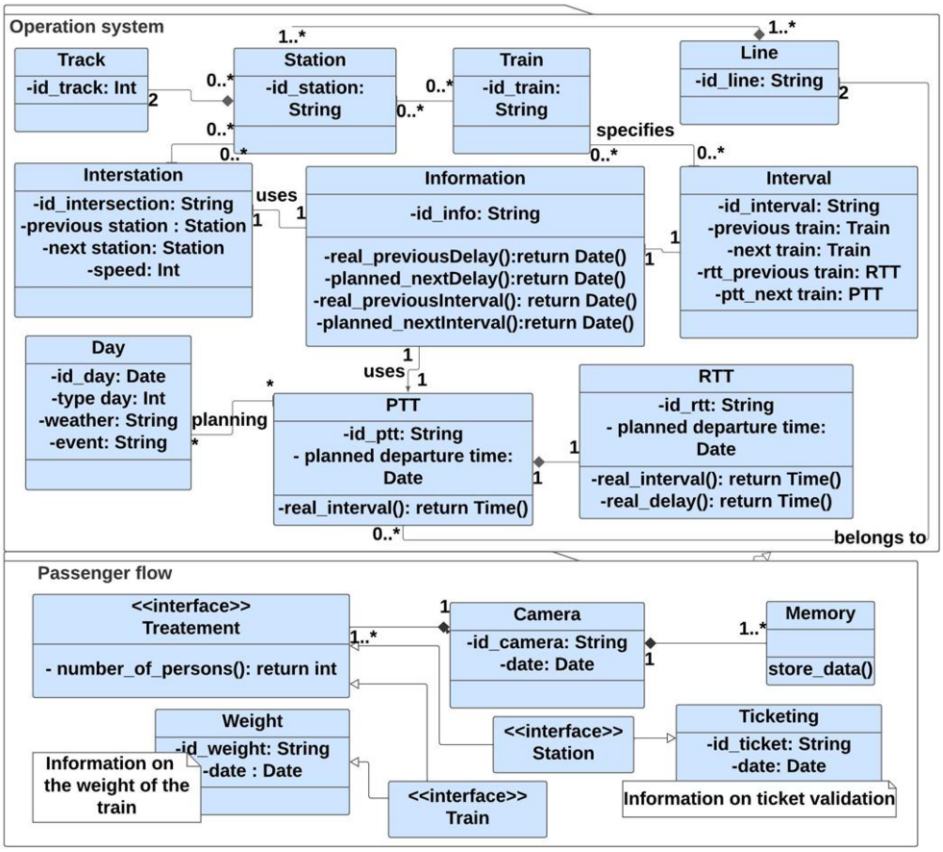

Figure 5. Final class diagram of operational and passenger data.

On the other hand, the "passenger flow" classes contain information related to the number of people on the platforms and trains, using systematic cameras, counting, and the weight of the train. This will help to better understand the passenger flow and anticipate any potential problems that may arise in case of high traffic. All the classes are linked to each other either directly or indirectly. For example, the station is linked to the train, track, interval, PTT, RTT, the passenger flow, and so on for each class. This method of work will allow for better management of railway traffic by facilitating the analysis and processing of data related to the operating system and passenger flow. By using separate packages, we can better understand the different aspects of the railway system and improve the efficiency of train management. Each operation in this diagram defines a set of mathematical operations performed to calculate variables such as actual and previous delays, intervals between consecutive trains. These links and operations facilitate the reading and understanding of railway data, making it easy to define essential data for each learning process (such as predicting delays and passenger flows for our next research project) and visualizing the data to help operators make the right decision in the control center.

In the next section, we present an example of our use case projected onto the class diagram presented in Figure 5.

4. Case study

The use of real-world data in data modeling is crucial to ensure the relevance and reliability of the results obtained. In this section, the actual train operating data used in was obtained from a collaboration with a railway control center that manages *n* stations and *m* trains. The raw data includes PTT, RTT and train numbers, station names, platforms, and the number of passengers per train. The calculated data includes current and previous train delays, as well as planned and real intervals. Some primary sample data is presented in this paper vaguely due to the intellectual property clauses promulgated by EGIS Rail. It is not possible to provide precise information about their sources or the details of the data itself. Nonetheless, the results obtained using this real data allowed for more robust and reliable data modeling, which helped to strengthen the validity of the conclusions drawn from this study. By using real data, models can be more easily adapted to real-world situations, leading to a better understanding of the phenomena studied and more relevant recommendations for policymakers and practitioners.

In this section, we provide an example based on our use case of a railway control center using raw database data. The "Use Case" section presents the characteristics of the studied railway line. Subsequently, the "Object diagram" section illustrates the transition from class diagram to data representation, the "Data Representation" section showcases the projection of this raw data onto the class diagram presented in Figure 5.

4.1. Use case

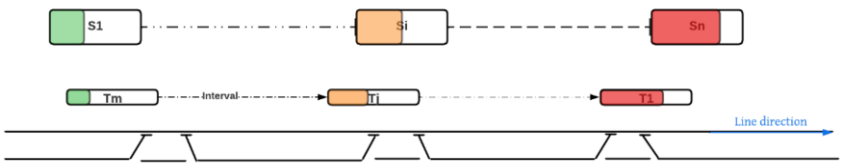

Figure 6. Schematic diagram demonstrating a subway line.

Figure 6 presents a comprehensive overview of the primary information for a subway line. This information includes *n* stations and *m* trains, the possible causes of disruption to the planned schedule, the number of passengers on the trains and on the platforms, this information is accompanied by a color-coded system that indicates the level of occupancy on both the metro trains and platforms, where green represents low occupancy, orange represents medium occupancy, and red represents high occupancy. Additionally, the interval indicating the distance maintained between two consecutive metro trains. The process of data modeling plays a vital role in facilitating the manipulation of all data variables and providing a clear visualization of their uses. In a simple subway line, there are two types of data: operational data and passenger load data us presented in Figure 5. Operational data are linked to the identification of the railway line, stations, subway, planned and real timetables. Passenger load data involves collecting data on the number of passengers on trains and platforms, using the train weight system. Relational modeling

is a widely used technique in database design and is particularly well-suited to handling complex data sets with many interrelated pieces of information. By breaking down a large, complex data set into smaller, more manageable tables and establishing relationships between them, relational modeling allows for more efficient querying and analysis of the data.

4.2. Object diagram

We had constructed our relational model in Figure 5, we projected the data from the control center presented in Table 1 onto the model. This allowed us to establish connections between the data and calculate any missing information using the operations defined in our classes.

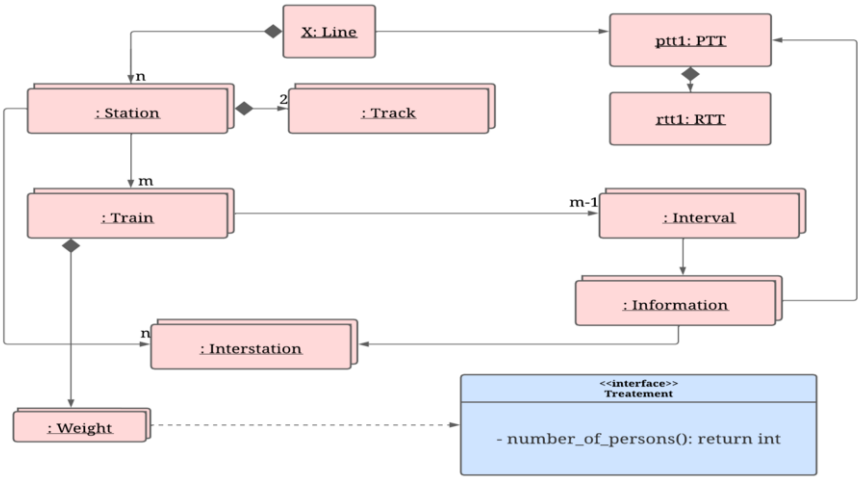

Figure 7. Object diagram representing the raw data.

The object diagram aims to connect the data presented in our "use case" to Tables 2, 3, and 4. By using the projected data from Table 1 on our class diagram presented in Figure 5, we can represent the raw data in an organized manner. In line X, there are *n* stations, each station having two tracks and *m* trains operating during a specific day, with planned timetable PTT in collaboration with control center operators. After projecting this data onto our class diagram, we obtain empty classes that need to be filled with calculation classes and methods. For instance, for each consecutive pair of trains, there is a dataset called Interval that is used to calculate the current and previous intervals, both planned and actual. Additionally, for each triplet of stations, there is an Interstation class dedicated to defining the preceding and subsequent stations for each train. Finally, there is a set of data calculated through the train weight interface, used to display the number of passengers per train, which is linked to each station.

4.3. Data representation

The raw data presented in Table 1 might contain missing or incomplete information. For instance, the table include the weight of the train, but it may not provide any information

about the number of passengers in trains. Similarly, the table may include information about the planned and real departure times of trains, but it may not provide any information about delays or real intervals between trains. To address these issues, we used the relational model presented at the top of this section. This model enabled us to identify new attributes presented in Table 2,3,4 and define the relationships between different variables. For example, using the model, using the Eq.2,3, we were able to calculate the planned and real intervals between two consecutive trains as shown in Table 2 with an example of subway data to calculate these intervals between *m* trains in Stations 3, which was not provided in the raw data. We were also able to calculate train delays using the Eq.1 by comparing the planned and real departure times of the trains shows in Table 3 with an example of subway data for Train number 2 in *n* Stations. Additionally, we were able to calculate the number of passengers on trains and platforms with Eq.4, Due to the confidentiality of the control center, we cannot present the details of this equation.

Table 1. Example of raw data for n stations and *m* trains obtained from the control center before using the relational model explained at the top of this section.

Planned departure time	Real departure time	Track	Train	Station	Weight of the train(ton)
12:05:00	12:05:56	1	01	$Station_1$	280
12:08:00	12:08:52	1	01	$Station_2$	1 75
…	…..	..	..	…	..
14:54:00	14:59:00	1	*m-1*	$Station_{n-1}$	130
15:00:00	15:03:33	1	*m*	$Station_n$	100

Table 2. Example of subway data for $Station_3$ and *m* trains after projecting the raw data presented in Table 1 onto the relational model described of this section.

Planned departure time	Real departure time	Track	Train	Delay(s)	Planned interval (s)	Real interval (s)	Number of people in platform
12:14:00	12:16:12	1	01	132	540	440	13
12:22:00	12:24:52	1	02	172	480	520	7
…	…..	..	..	..	..	…	..
14:04:00	14:03:19	1	*m*	-41	240	252	20

Table 3. Example of subway data for Train number 2 in *n* stations after projecting the raw data presented in Table 1 onto the relational model described of this section.

Planned departure time	Real departure time	Track	Station	Delay (s)	Planned interval (s)	Real interval (s)	Number of people in platform
12:22:00	12:24:52	1	$Station_3$	172	480	520	7
12:22:00	12:26:15	1	$Station_4$	255	480	512	10
…	…..	..	…	..	..	…	..
12:43:00	12:48:37	1	Station_n	337	300	332	20

Table 4. Example of subway data for *n* stations and *m* trains after projecting the raw data presented in Table 1 onto the relational model described of this section.

Planned departure time	Real departure time	Track	Train	Station	Delay (s)	Planned interval (s)	Real interval (s)	Number of people in platform
12:10:00	12:10:26	1	01	$Station_3$	26	0	0	10
12:14:00	12:14:05	1	02	$Station_1$	5	540	489	35
…	…..	..	..	…				
14:54:00	14:59:00	1	*m-1*	$Station_{n-1}$	300	300	344	30
15:00:00	15:03:33	1	*m*	$Station_n$	153	300	186	20

We consider an interest train t_j that has just arrived at station s_i, where $s_{ij} \in \{s_{1j}, s_{2j}, \dots, s_{nj}\}$. The problem is to calculate the departure delay and expected interval of t_j at station s_i such that t_{j-1} is the preceding train. To calculate the delays and intervals using the methods defined in our class diagram, we rely on the projection of raw data presented in Table 1 onto the class diagram to obtain the calculable data presented in Table 4. The information of interest trains from station s_{1j} to station s_{nj} includes operational information (Timetables) of the trains, calculated features defined as follows:

X1: Planned departure time in P_{ij} *X2: Real departure time in* R_{ij}

X3: Track (0,1) *X4: Name of station* s_{ij}

$$X5: Delay(s_{ij}) = R_{ij} - P_{ij} \ in\ seconds \quad (1)$$

$$X6: Planned\ interval\ (s_{ij}) = P_{ij} - P_{i(j-1)}\ in\ seconds \quad (2)$$

$$X7: Real\ \ interval\ (s_{ij}) = R_{ij} - R_{i(j-1)}\ in\ seconds \quad (3)$$

$$X8: Number\ of\ people\ (s_i) = \blacksquare\blacksquare\blacksquare\blacksquare\blacksquare\blacksquare \quad (4)$$

The relational model proposed in this article provides a clearer understanding of the relationships within a railway line, as illustrated in our specific use case. It demonstrates the strength of our model by indirectly impacting working time. Instead of randomly selecting data for each learning problem, such as delays and passenger flow, we can directly identify the relevant connections and variables for targeted predictions. Additionally, the proposed methods in the diagram allow us to calculate missing variables, such as delays, intervals, and passenger flows, using the available raw data. This model serves as a foundation for the next phase, the hybrid approach, which combines machine learning with the steps outlined in Figure 1. The aim is to develop a decision support system that regulates delays based on passenger flow. By leveraging the relational model, we can optimize the prediction process and enhance the overall efficiency of the railway system.

5. Conclusion

Railway data modeling is a crucial step in developing machine learning algorithms to predict delays and passenger flows. The use of actual railroad operating data, as well as collaboration with railroad control centers, provides accurate and comprehensive data

for modeling. This process involves obtaining and analyzing critical information such as planned and actual departure dates, train numbers, station names, platforms, and passenger counts. With this data, the next step is to develop machine learning algorithms to predict delays and passenger flows, providing valuable insights for rail operators to make informed decisions.

As the field of machine learning continues to evolve, data modeling of rail systems becomes increasingly important in creating more efficient and reliable transportation systems. The utilization of our relational model in the hybrid approach has helped us select the variables relevant to each classification problem, such as delays and passenger flows. This enables us to preprocess the chosen data, structure it, and prepare it for each prediction algorithm. The prediction algorithms are chosen based on their suitability. In our case, we start by testing time series algorithms as there is a dependence between data points. We optimize their parameters to achieve high performance in predictions. We then compare these models with other machine learning models to select the most suitable algorithms for our situation. We subsequently generate real-time calculations to validate the predictions at time t using calculations in previous stations.

References

[1] A. Stathopoulos et M. G. Karlaftis, « A multivariate state space approach for urban traffic flow modeling and prediction », *Transp. Res. Part C Emerg. Technol.*, vol. 11, nº 2, p. 121-135, avr. 2003, doi: 10.1016/S0968-090X(03)00004-4.

[2] H. Flier, R. Gelashvili, T. Graffagnino, et M. Nunkesser, « Mining Railway Delay Dependencies in Large-Scale Real-World Delay Data », in *Robust and Online Large-Scale Optimization*, R. K. Ahuja, R. H. Möhring, et C. D. Zaroliagis, Éd., in Lecture Notes in Computer Science, vol. 5868. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009, p. 354-368. doi: 10.1007/978-3-642-05465-5_15.

[3] F. Corman et L. Meng, « A Review of Online Dynamic Models and Algorithms for Railway Traffic Management », *IEEE Trans. Intell. Transp. Syst.*, vol. 16, nº 3, p. 1274-1284, juin 2015, doi: 10.1109/TITS.2014.2358392.

[4] D. G. Bearfield, « Causal Modelling of Lower Consequence Rail Safety Incidents ».

[5] N. Marković, S. Milinković, K. S. Tikhonov, et P. Schonfeld, « Analyzing passenger train arrival delays with support vector regression », *Transp. Res. Part C Emerg. Technol.*, vol. 56, p. 251-262, juill. 2015, doi: 10.1016/j.trc.2015.04.004.

[6] A. Berger, A. Gebhardt, M. Müller-Hannemann, et M. Ostrowski, « Stochastic Delay Prediction in Large Train Networks », p. 12 pages, 2011, doi: 10.4230/OASICS.ATMOS.2011.100.

[7] X. Bao, Y. Li, J. Li, R. Shi, et X. Ding, « Prediction of Train Arrival Delay Using Hybrid ELM-PSO Approach », *J. Adv. Transp.*, vol. 2021, p. 1-15, juin 2021, doi: 10.1155/2021/7763126.

[8] R. Shi, X. Xu, J. Li, et Y. Li, « Prediction and analysis of train arrival delay based on XGBoost and Bayesian optimization », *Appl. Soft Comput.*, vol. 109, p. 107538, sept. 2021, doi: 10.1016/j.asoc.2021.107538.

[9] F. Corman et E. Quaglietta, « Closing the loop in real-time railway control: Framework design and impacts on operations », *Transp. Res. Part C Emerg. Technol.*, vol. 54, p. 15-39, mai 2015, doi: 10.1016/j.trc.2015.01.014.

[10] C. Dong, C. Shao, S. H. Richards, et L. D. Han, « Flow rate and time mean speed predictions for the urban freeway network using state space models », *Transp. Res. Part C Emerg. Technol.*, vol. 43, p. 20-32, juin 2014, doi: 10.1016/j.trc.2014.02.014.

[11] Z. Jiang, C.-H. Hsu, D. Zhang, et X. Zou, « Evaluating rail transit timetable using big passengers' data », *J. Comput. Syst. Sci.*, vol. 82, nº 1, p. 144-155, févr. 2016, doi: 10.1016/j.jcss.2015.08.004.

[12] M. Yaghini, M. M. Khoshraftar, et M. Seyedabadi, « Railway passenger train delay prediction via neural network model: RAILWAY PASSENGER TRAIN DELAY PREDICTION », *J. Adv. Transp.*, vol. 47, nº 3, p. 355-368, avr. 2013, doi: 10.1002/atr.193.

[13] Olga Sadouskaya, MD, « How to Predict the Weather: The Science Behind Weather Forecasting», apr. 2023.

[14] D. Berardi, D. Calvanese, et G. De Giacomo, « Reasoning on UML class diagrams », Artif. Intell., vol. 168, no 1-2, p. 70-118, oct. 2005, doi: 10.1016/j.artint.2005.05.003.

[15] W. Mou, Z. Cheng, et C. Wen, « Predictive Model of Train Delays in a Railway System », 2019.