energies

Article

Energetic Map Data Imputation: A Machine Learning Approach

Tobias Straub [1,*], Mandy Nagy [2], Maxim Sidorov [3], Leonardo Tonetto [2], Michael Frey [1] and Frank Gauterin [1]

[1] Institute of Vehicle System Technology, Karlsruhe Institute of Technology (KIT), 76131 Karlsruhe, Germany; michael.frey@kit.edu (M.F.); frank.gauterin@kit.edu (F.G.)
[2] Department of Informatics, Technical University of Munich, 85748 Garching, Germany; madalinamandy.nagy@tum.de (M.N.); tonetto@in.tum.de (L.T.)
[3] BMW Group, 80788 Munich, Germany; maxim.sidorov@bmw.de
[*] Correspondence: tobias.straub@partner.kit.edu; Tel.: +49-89-382-76090

Received: 13 January 2020; Accepted: 19 February 2020; Published: 22 February 2020

Abstract: Despite a rapid increase of public interest for electric mobility, several factors still impede Battery Electric Vehicles' (BEVs) acceptance. These factors include their limited range and inconvenient charging. For mitigating these limitations to users, certain BEV-specific services are required. Therefore, such services provide a reliable range prediction and routing, including charging-stop planning. The basis of these services is a precise and reliable Energy Demand (ED) prediction. For that matter, aggregated fleet-vehicle data combined with map-specific data (e.g., road slope) form an energetic map, which can serve for precise ED predictions. However, data coverage is paramount for these predictions, more specifically regarding gapless energetic maps. This work aims to eliminate the energetic map's gaps using two Machine Learning (ML) approaches: regression and classification. The proposed ML solution builds upon the synergy between map-information and crowdsourced driving profiles of 4.6 million kilometres of training and test traces. For evaluation, two test-scenarios capture the models' performance for the analysed problem in two perspectives. First, we evaluate our ML models, followed by the problem-specific energetic evaluation perspective for better interpretability. From the latter, the results indicate energetic map data imputation performs promisingly better when using the regression instead of the classification model.

Keywords: electric mobility; big data; artificial intelligence; supervised machine learning; regression; classification; missing data imputation

1. Introduction

Environmental awareness in both society and legislation demand a reduction of greenhouse gas emissions. This desire fosters interest in electrified personal mobility. In contrast to established fuel-based vehicles, battery electric vehicles do not emit greenhouse gases while driving. Nevertheless, today's electric vehicles have a rather limited range. The limited range is among the statistically significant ($p < 0.01$) factors decreasing the acceptance of electric vehicles, besides high prices and issues related to charging [1,2]. All these factors inhibit potential buyers from purchasing electric vehicles [1]. However, increasing the range of vehicles by installing a larger battery capacity is not an appropriate solution from different points of view. Accordingly, the already high price would further increase, demanding additional resources from an ecological viewpoint. Therefore, an alternative solution is required to lower the impact of the named impediments. Such services as precise range estimation and BEV route planning can achieve the required acceptance gain [2]. These services help

BEVs' users to utilise the available capacity entirely. Nevertheless, to strengthen the user's trust in those services, they need to operate reliably.

For reliable results, such services require detailed route-specific information about the prospective vehicle's Energy Demand (ED). A route's ED depends on multiple factors, including vehicle characteristics (e.g., vehicle mass and aerodynamic properties), road slope profile, traffic condition, weather and driving profile. A significant influence on the driving profile, and hence on the ED, is the driver's behaviour.

Different approaches in literature aim to predict the ED along an entire route, thus up to several hundreds of kilometres. These approaches need to model the mentioned influences, which refer to the categories of driving and slope profile, as well as surrounding parameters like temperature, atmospheric pressure and solar radiation. Those are the inputs for models which calculate the aimed output of the predicted ED. According to the literature review, both the model itself and the origin of the inputs are essential and can be utilised for grouping the route ED prediction models. Concerning the models, there are direct, machine learning and analytical models. Direct modelling is a simplistic approach that uses the recorded ED on a road section for prediction, as described, for instance, by Sellschopp [3]. Lamprecht's direct approach collects consumption-related information over time for each road class and normalises and denormalises it to and from driving on a road with no slope, respectively [4]. However, this approach can hardly describe variations caused by some of the influences mentioned above, such as traffic or temperature. Therefore, e.g., Masikos et al. tackles this drawback by using a general regression Neuronal Network model while using the ED model to generate routes with low ED [5]. With regard to the model types, as mentioned earlier, this is a direct model augmented by a machine learning algorithm. There are also pure machine learning models, which use different regression models, to predict ED [6], range [7] or to create routes with lower ED [8]. Nevertheless, machine learning models lack physical traceability of a change in the input variables and their impact on ED. In contrast, analytical models like proposed by Yang et al. and Wang et al. calculate the ED based on the driving resistance equation from a predicted driving profile [9,10]. This approach ensures a physically proper consideration of energetic influences and comes with straightforward physical traceability of the model. However, the model of Yang et al. uses only one overall efficiency, which could cause energetic deviations while recuperation phases [9]. Concerning the model type, analytic models are the most advantageous since they can correctly incorporate influences and at the same time make it physically traceable.

Regarding the input origins, the information mainly come from two sources: map data and vehicle data. Depending on the model type itself, the models based on map data either generate a driving profile from the map features, such as speed limit or crossings, and feed it to an analytical ED model [8,10] or directly predict ED like Masikos et al., which use a general regression Neuronal Network model [5]. In contrast to the map data-based model group, the vehicle data one consists of two subgroups: single vehicle and fleet-based data sourcing. The approach of Qi et al. only uses vehicle data recorded from itself, which reduces the area of prediction to routes already driven by the same vehicle [6]. However, Qi et al. also suggests transferring the data sourcing model to a fleet of vehicles, which in turn would enable the prediction of unknown routes [6]. The model of Grubwinkler et al. is based on fleet data from one urban area with high population density but ignores some roads in the prediction phase [7].

Apparently, a concept that works reliably in any area and can deal with different daytimes and weekdays, for modelling of rush hour effects, needs some backup mechanisms to tackle such issues as loss of connection or absence of data. Therefore, an imputation of missing data can enhance the robustness of the ED prediction systems. Generally, the prediction of missing data is an essential area in multiple, distinct domains [11–14]. Regardless of the application field, simply ignoring and discarding the data is not recommended [11,15]. Thus, the estimation of missing information utilises approaches of different complexity levels, from the naive imputation of the mean to more sophisticated predictive models based on ML. In the area of psychological counselling, Schlomer et al.

compares the different performances achieved by software that implements statistical methods for data imputation [11]. At this level, multiple imputation methods and full information maximum likelihood provide better results than the mean substitution methods, which are not recommended for further usage. Besides statistical approaches, the usage of ML methods is analysed in medicine [12], industrial data management [16] and transportation [14]. Jerez et al. make use of both statistical imputation and ML-based strategies, and conclude that the latter provides higher prediction accuracy [12]. Despite the popularity of these two main imputation methods, Jeng et al. introduce a domain scenario in which the application of sophisticated models might be expensive in both economic and resource dimensions; therefore, two functional approaches corresponding to online and offline processing are developed [13].

In the research of traffic forecasting, the representative patent of Wynter et al. shows usage of historical data as input to an exemplary missing data prediction algorithm [14] . Similarly, Laña et al. present ML approaches to impute lacking traffic information, based on regression and classification models [17]. Although most of the complex imputation problems are searching for solutions in ML methodology, the dimension of ML opens structural and parametric optimisation problems, also addressed in [18,19].

Missing data imputation for the ED prediction of BEVs has a lower presence in the literature. Different approaches are based on own vehicle data suggest imputing data from other vehicles in case of missing data [6,20]. Accordingly, a fleet-vehicle data-based one has the advantage that it can naturally reflect effects on ED at a location, whereas a map-based approach needs to reconstruct the effects. To summarise, this paper aims to present a backup mechanism for situations without sufficient data which works within an analytical ED prediction based on fleet data. Whereby, such location-specific fleet data within this ED prediction context provide a map layer, which can be figured as an energetic map with some gaps. The main objective of this work is to present a concept for such a favourable ED prediction based on fleet data. Moreover, as a proof of concept, the prediction also features a backup mechanism to deal with an insufficient amount of data, which can be understood as an energetic map data imputation.

2. Materials and Methods

The main objective of the Energy demand Prediction Framework (EPF) is to raise the acceptance level of BEVs among customers by enabling functions supporting the driver. Therefore, the model must ensure a robust and precise prediction output. In this context, the identification of error sources becomes essential. Missing data is among the contributors to the final prediction error. This section aims to introduce the fleet data-based EPF and the problem of missing data, by describing the methods involved in the proposed solution and presenting the evaluation setup from two different perspectives.

2.1. Fleet Data-Based Route Energy Demand Prediction

Before describing the actual problem addressed by this paper, the following section first explains the general description of a route's ED and then the EPF based on it. Equation (1) depicts the calculation of a routes ED at the high voltage battery's (HVB) clamp $E_{\text{Route, HVB, Cl}}$, which is the location of the connection to all energy consumers. It is the sum of the powertrain's $E_{Pt,[k]}$ and auxiliaries' energy demand $E_{Aux,[k]}$ on each route link k from the start k_{start} to the destination link k_{dest}, where links represent a road segments located between neighbouring road junctions. In the following, it is assumed that the problem is quasi-static along a road segments. In other words, the auxiliaries' power demand $P_{Aux,[k]}$ is nearly constant within a link. As the map provides each link's length $\Delta s_{[k]}$, and the average velocity $\bar{v}_{[k]}$ (AV) is also available, the estimated time to drive along a link can be calculated. The product of this value and the $P_{Aux,[k]}$ results in each link's $E_{Aux,[k]}$. In contrast, the calculation of the other part of the ED on a link, the $E_{Pt,[k]}$, is not time-dependent, but distance-dependent. Accordingly, the $E_{Pt,[k]}$ is the integral of wheel force F_W concerning the driven distance s, given the powertrain's efficiency η_{Pt} is considered according to the direction of F_W. Whereby, the F_W indicates whether there is ED from the HVB and the powertrain is in propulsion (prop) mode or energy can be recovered into the HVB, and the powertrain is in recuperation (recu) mode.

$$E_{\text{Route, HVB, Cl}} = \sum_{k=k_{start}}^{k_{dest}} E_{Pt,[k]} + E_{Aux,[k]} = \sum_{k=k_{start}}^{k_{dest}} \int_{k}^{k+1} \frac{1}{\eta_{Pt}^{sign(F_W(s))}} \cdot F_W(s)ds + P_{Aux,[k]} \cdot \frac{\Delta s_{[k]}}{\bar{v}_{[k]}} \quad (1)$$

$$F_W(s) = \vec{p}^T \cdot \vec{r} = \begin{pmatrix} m \cdot e \\ \frac{\rho}{2} \cdot c_x \cdot A \\ f_t \cdot m \cdot g \\ m \cdot g \end{pmatrix}^T \cdot \begin{pmatrix} a(s) \\ v^2(s) \\ \cos(\alpha) \\ sin(\alpha) \end{pmatrix} = F_{inertia} + F_{air} + F_{tire} + F_{slope} \quad (2)$$

Equation (2) depicts the calculation of F_W. As aforementioned, it separates the quasi-static parts of mainly vehicle parameters in the parameter vector $\vec{p}$ (vehicle mass m, gravitational acceleration g, rotational mass factor e, tire rolling resistance coefficient f_t, air density ρ, aerodynamic drag coefficient c_x and frontal area A) and the variables changing over distance in the route vector $\vec{r}$. The $\vec{r}$ contains the slope α and the DP, which consists of the velocity v and the acceleration a, over distace s. The vector product of those two vectors embody the sum of driving resistance forces, namely, the inertial pseudo d'Alembert force $F_{inertia}$, the aerodynamic resistance force F_{air}, the tires' rolling resistance force F_{tire} and the climbing resistance force F_{slope}. Table 1 shows the energetic driving profile map attributes (DPMAs) that can be extracted if only looking at the DP influence from the Equations (1) and (2). Note that the integral DPMA need to be separated according to the energy's flow into propulsion and recuperation parts to consider η_{Pt} properly.

Table 1. Derived energetic driving profile map attributes (DPMAs), which generally describe the impact of the driving profile on a vehicle's energy demand.

Driving Profile Map Attribute	Propulsion	Recuperation
Integral Acceleration	IAP: $\int [a(s)]_{Prop} ds$	IAR: $\int [a(s)]_{Recu} ds$
Integral Squared Velocity	ISVP: $\int [v^2(s)]_{Prop} ds$	ISVR: $\int [v^2(s)]_{Recu} ds$
Average Velocity	AV: $\bar{v}$	

Figure 1 describes the process of the EPF and its intermediate stages from (A) to (I) based on the derived DPMA. The figure shows in two boxes which parts of the process take place in vehicles and on backend servers. For the first process step (A), the vehicles of a fleet record traces. A trace includes second-wise records of timestamps and the vehicle's geolocation retrieved from a Global Navigation Satellite System (GNSS). The vehicle sends those records anonymised via a mobile network to a backend server, as indicated by step (B). In step (C), the backend performs map matching, which adds to every record a map-link reference, called a link-ID. The map matching provides each record's relative position on the link. The used map matching is based on a Hidden Markov Model as described by [21].

Furthermore, the link-ID enables the addition of other map attributes regarding each record. This attributes can include the link's length, the speed limit, the direction of movement on the link and others. In particular, based on each link's lengths together with the relative position on the link and the timestamp, the backend part of the framework can calculate a continuous reconstruction of the DP in step (D). This DP describes the velocity and acceleration over the driven distance. From this DP, the following step (E) extracts energetic DPMA for crossed map-links. Furthermore, the EPF's proof of concept incorporates a discretization of the map-links into multiple sublinks of maximum 50 m length. Accordingly, step (E) calculates the DPMA for each sublink traversed by each trace. Subsequently, the five aggregated DPMA are a sample of the influence of this trace's DP on ED along a sublink, as described in the beginning of this section and summarised in Table 1. Step (F) aggregates DPMA of different traces into nonparametric distributions for distinct space and time buckets for each DPMA, named DP map attribute distributions (DPMADs). Specifically, on one hand, a space bucket refers to the particular sublink of a record and the direction of driving on it. On the other hand, a time bucket refers to the corresponding weekday and the number of half-hours elapsed from midnight until the record of the DPMAs. Therefore, a group of five DPMADs refers to DPMA samples. This means that the samples belong to similar circumstances considering time and location. Consequently, a DPMAD describes the spread of traces concerning the ED influence of the DP. In turn, the DPMADs represent the spread in cumulated probability within 25 bins for each velocity-related DPMA, whereas 31 bins represent the acceleration-based ones. Therefore, the population of the DPMADs forms a map of the energetic spread stored on the backend.

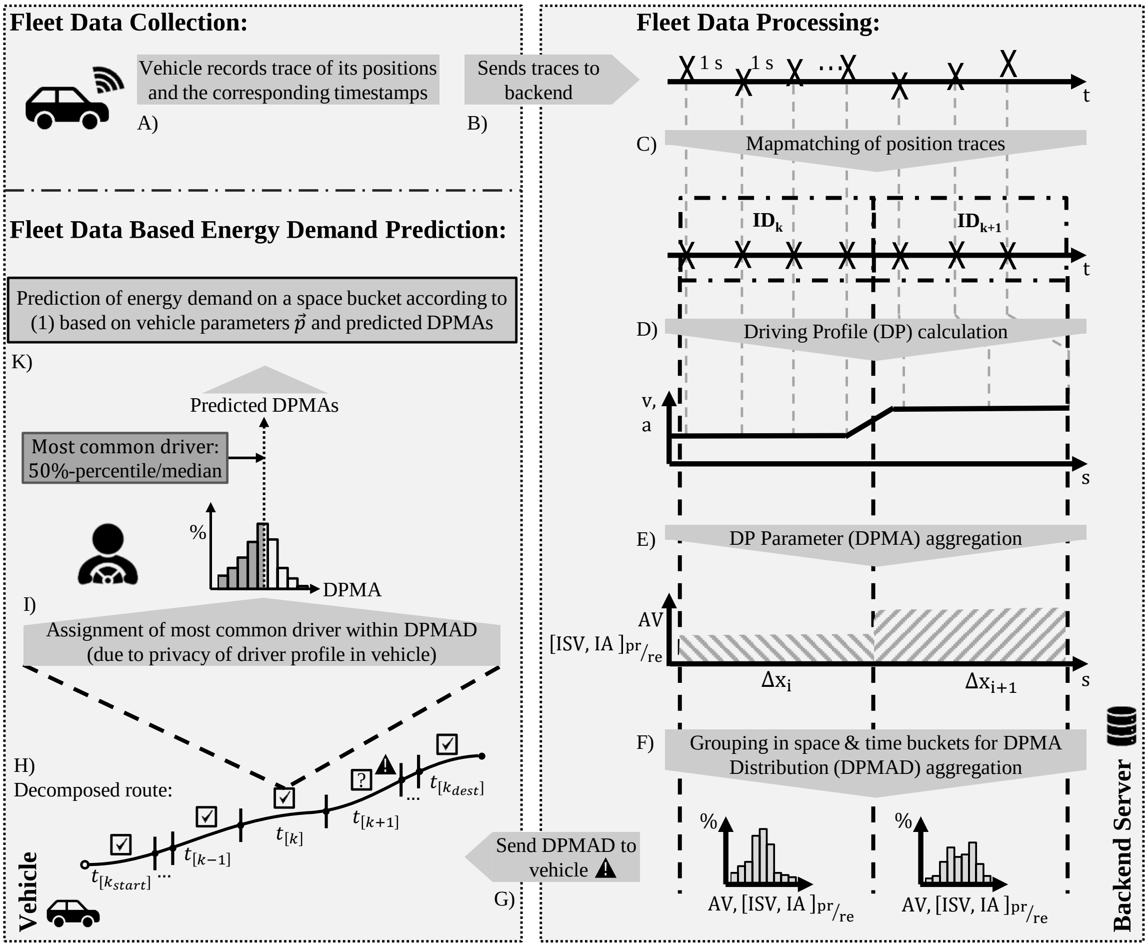
Fleet Data Collection:
Vehicle records trace of its positions and the corresponding timestamps
A)
Sends traces to backend
B)
Fleet Data Processing:
1 s
1 s
t
C)
Mapmatching of position traces
ID_k
ID_k+1
t
D)
Driving Profile (DP) calculation
v, a
s
E)
DP Parameter (DPMA) aggregation
AV
[ISV, IA]pr/re
Δx_i
Δx_i+1
s
F)
Grouping in space & time buckets for DPMA Distribution (DPMAD) aggregation
%
AV, [ISV, IA]pr/re
%
AV, [ISV, IA]pr/re
Backend Server
Fleet Data Based Energy Demand Prediction:
Prediction of energy demand on a space bucket according to (1) based on vehicle parameters p⃗ and predicted DPMAs
K)
Predicted DPMAs
Most common driver: 50%-percentile/median
%
DPMA
I)
Assignment of most common driver within DPMAD (due to privacy of driver profile in vehicle)
H)
Decomposed route:
t[k_start]
t[k−1]
t[k]
t[k+1]
t[k_dest]
Send DPMAD to vehicle
G)
Vehicle

Figure 1. The general approach of fleet-based route energy demand prediction framework (EPF). The process starts from fleet data collection (A) in a vehicle and sending to backend server (B). It continues with the processing in the backend in the steps (C)–(F). After sending (G) the DPMA to the prediction vehicle, the ED prediction (K) takes place for the route that is divided into space buckets. Steps marked with warning signs are possible faulty and can fail in this process. In step (G), the connection to the vehicle can be lost. In the decomposed route (H), some space buckets for the needed time bucket can be missing due to no or not sufficient fleet data recordings.

More specifically, when a vehicle plans an upcoming route, the backend sends in step (G) the recorded DPMADs to the vehicle, which correspond to the required time and space bucket. Therefore, the map referenced DPMADs can be figured as an energetic map.

Due to data privacy reasons, the following steps (H)–(K) take place in the vehicle. Particularly, step (H) extracts driver-specific DPMA out of each DPMAD based on previously learned driver-behaviour-profile. This approach ensures data privacy by architectural setup. Potentially, step (H) can use a ranking of the driver's energetic behaviour within the recorded spread from the fleet. As DPMADs are frequently non-Gaussian or bimodal distributed, percentile values are the basis for the ranking. For such cases, a standard deviation-based ranking is not applicable. This driver ranking ensures a personalised ED prediction by utilising features from a driver's behaviour as input to the model. As this work aims to establish a single solution throughout the fleet, the following evaluations use the most frequent or median behaviour recorded using the crowdsourcing approach, which refers to the 50th percentile.

Once the DPMAs are extracted from the DPMADs, the last step (I) performs the actual energy demand prediction. The prediction step calculates the energy demand with regard to (1) and (2) on each sublink along the route and eventually sums them up. The overall process of the EPF, shown in Figure 1, also highlights two steps, which can be faulty in the context of a backend-based fleet data concept. The latter can happen either in step (F), where the connection between vehicle and backend could be lost, or in step (H), where specific DPMAD for certain space-time buckets could not be available.

2.2. Principe of Machine Learning Approach

As depicted above, the process of energy prediction relies on the data collected from the fleet. Moreover, the transmitted information links the model and available vehicle data. Naturally, the fleet coverage directly affects the prediction performance. If the recording vehicles did not cover a specific time bucket on a sublink, the model should cope with gaps while predicting the ED.

The proposed solution utilises an ML approach using both map information and existing fleet data. Figure 2 shows the overall architecture of the proposed solution in the form of a blackbox representation. More precisely, the missing data identification layer aims to detect faulty situations also described in step (H) of Figure 1, to find sublinks with missing fleet information. Subsequently, the input layer depicts the features. Specifically, the features comprise map-related information, such as road-specific attributes described in more details in the following Section 2.3. Based on map-link IDs, the map information and available DPMADs received from the fleet are aligned for training. After the input layer, the blackbox layer outlines the proposed solution approaches, including the suggested ML model types, which should predict the DPMAD information in the output layer. On one hand, the regression model can directly predict the bins of the DPMADs, as detailed in Section 2.4. On the other hand, the classification type model can only predict categories of DPMADs, which a clustering provides in the shape of representative DPMADs, as depicted in Section 2.5.

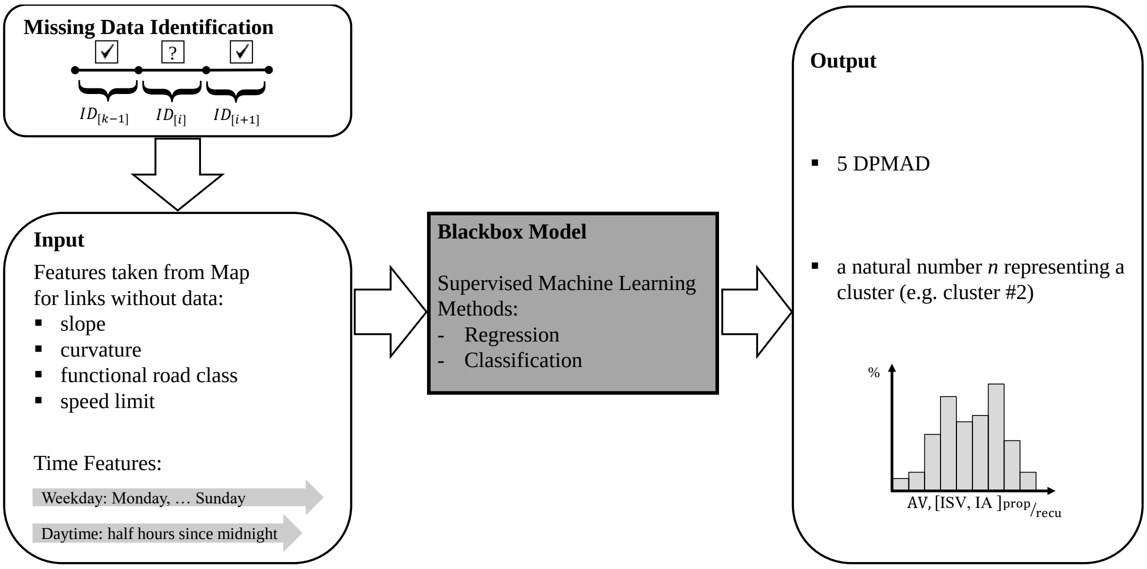

Figure 2. Blackbox architecture.

To assure a comprehensive explanation of the blackbox layer, first, the principle of Machine Learning is introduced. As formally defined by Mitchell, the learning performance of a system increases based on an experience generalisation process on a given task [22]. Based on the experience dimension and the format of training data provided to the model, the learning process splits into supervised, semisupervised, and unsupervised ones. Given the first category, the goal is to learn the prediction of a random variable by relying on a set of explicative random variables. The input data for training the ML model, in this case, is labelled, constructing an alignment between input and output structures. In contrast, the unsupervised learning aims to learn the probability distribution of a random vector while the training data is unlabelled. It is mainly the right approach when the problem can not be formalised as a blackbox, and the scope is to learn unknown patterns of the data. In turn, semisupervised learning involves both types of training data, where the presence of labelled information in combination with unlabelled data provides a considerable performance boost compared to unsupervised learning. The addressed missing fleet data problem fits in the supervised ML context. Indeed, the problem is well defined, and the map-link IDs together with timestamps provide matching between the input of map information, and the output of DPMADs collected from the vehicles in space and time.

The ability to manipulate large amounts of data is the result of a tight combination between the algorithms upon which ML builds and the resources requested in terms of computational power. The limits of application areas over which ML can action are continuously shrinking. As a consequence, in the field of energy systems, the use of ML models have recently met ample growth [23–25]. In the case of supervised ML, the main problems addressed are tasks belonging to two categories: regression and classification. The proposed solution includes an approach for both categories, and the objective is a comparison between the performances given two central perspectives: a general, more mathematic ML perspective (MLP) and a problem-specific energetic perspective (EP). The algorithm used for the regression method is the Linear Regression Section 2.4, and for classification is the Decision Trees algorithm Section 2.5. In terms of complexity, these are simple, fast and easy to scale algorithms, qualifying them for the presented proof of concept. Moreover, the benefit of these algorithms lies in their position as representatives of the regression and classification methods. On a system level, they are subject to change similar to their parameters.

2.3. Feature Selection

Prior to setting the algorithm parameters of the proposed regression and classification models, the input features for the supervised ML approach need to be carefully selected. Additionally, the feature selection needs to consider the domain-specific influences. Therefore, Equations (1) and (2) describe such relevant influences on a vehicle's ED. Therefore, these equations are the basis for the selection of the relevant features from the available map attributes. In the following, these attributes describing the roads of a road network are called map features. As mentioned in Section 2.1, the route vector $\vec{r}$ denotes the energetic influence of the route being driven. This vector includes the topographical and DP factors. The digital map represents the topographical characteristics by spatial shape points, forming map-links or sublinks, along with corresponding slope angles to a horizontal surface. However, the aimed supervised ML in a proof of concept requires scalar features. Accordingly, aggregating functions need to describe slope values along sublinks, in a way that it not only summarises but also considers extreme values. Therefore, average, maximum and minimum slope of a sublink serve as features of the topographical influence on ED. These features can capture the topological influence on ED, e.g., in the case of a high negative slope value equal to going downhill, which leads to a lower ED or to the possibility to charge the battery by recuperating potential energy and vice versa.

As Equation (2) depicts, the other essential part of the route vector $\vec{r}$ is the DP, comprising two components: velocity and acceleration. Multiple features model these two components within our framework. As of velocity, which influences the ED by means of aerodynamic resistance, the framework uses the map attribute speed limit, as a primary feature for this component. However, traffic and road characteristics are also influencing the actual velocity. Therefore, the further features of daytime and weekday with a resolution of 30 min model the traffic influence in a first iteration. In this proof of concept, these features enable the ML to consider effects of rush hour traffic. Apparently, traffic influences not only the velocity, but also acceleration; therefore, we used this time-specific feature to address both velocity and acceleration DP components, which affect the ED due to a change in aerodynamic resistance and a change in the vehicle's kinetic energy.

Similarly, road class and curvature influence both parts of the energetic DP, namely, velocity and acceleration. The road class feature represents road characteristics like speed, the volume of traffic and geographical position, including the importance of areas which are connected through the roads. The road classes' definition is categorical, and the data used within the proposed system use five road class categories. These categories start from highest traversal roads connecting significant urban areas to roads with a lower traffic volume and minimum speed limits.

Consequently, the road class describes the character of the road and thus affects typical velocity and acceleration being registered along the route, which again influences the ED due to a change in aerodynamic resistance and a change in the vehicle's kinetic energy. In turn, the curvature affects the DP capturing the physical and comfort restrictions given by the road geometry regarding the curves' radius. Thus, given a particular curvature, only a certain velocity can be driven comfortably and securely. Similar to slopes, also for curvatures, the map provides multiple values along a sublink. Correspondingly, aggregating functions extract two features from the curvature values along a sublink: minimum and maximum.

The selected map features geometrically describe a map. The basic units of this map are elements called links and nodes, with links combining shape points organising roads in a continuous way and nodes represent crossing by connecting several links. The shape points encode the real-world curvature of the road into measured numerical values. The nodes also determine the direction within a link, therefore, depending on the driving direction can be either Origin or Non-Origin nodes.

Note that the selected features are subject to optimisation, as we are considering them as a system parameter. Furthermore, feature engineering techniques can be applied, for instance, polynomial combinations of the features with a higher degree. Finally, more sophisticated features like weather, real-time traffic information or the speed limit difference between adjacent links can be included in the model.

In the blackbox representation of the presented framework, the output block contains five DPMADs built upon the historical DPMADs captured from the fleet, as explained in Section 2.1. Figure 3 shows representatives of these DPMADs.

2.4. Application of Regression

Once the input features are selected, a suitable algorithm for regression problem has to be chosen. The most commonly used one is Least Squares. Fundamentally, it determines the decision boundary line for a set of data points by minimising the sum of the squares of the errors between the hypothesis of the model and the observed values. When compared with other regression methods, results of Least Squares are easily interpretable and can be achieved in a smaller computational effort (i.e., faster execution times and lower memory footprint). This method, however, is highly sensitive to the choice of starting values for its coefficients and well known for not capturing highly complex relationships between the input features [26]. Moreover, this method has a closed-form solution which can be computed using linear algebra given a feasible dimension of the input vector. In the supervised ML method integrated into this work, the Linear Regression algorithm is used as a prediction model. Its default parameters are defined in Table 2.

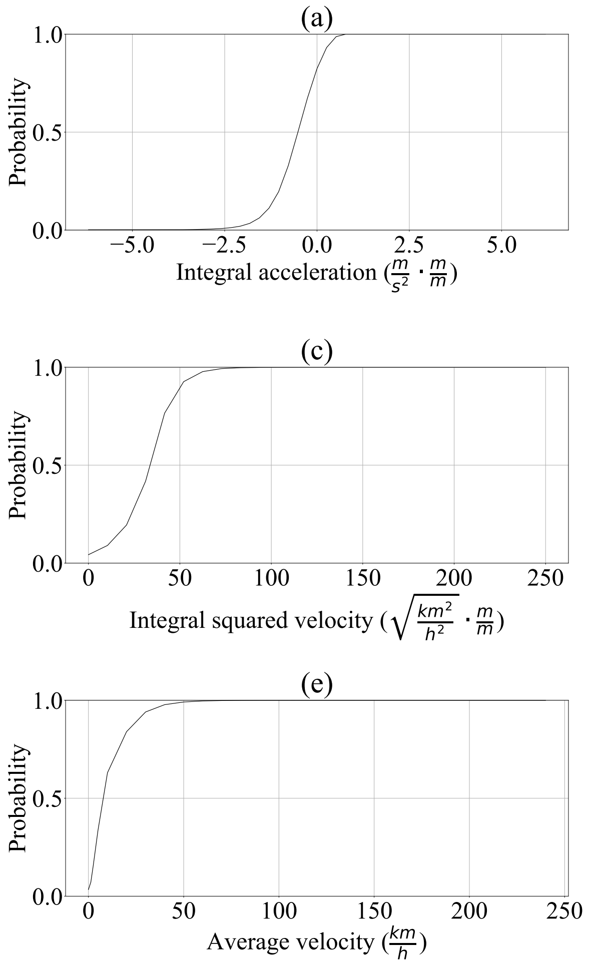

Figure 3. Representative driving profile (DP) map attribute distributions (DPMADs): (**a**) Integral acceleration recuperation (IAR). (**b**) Integral acceleration propulsion (IAP). (**c**) Integral squared velocity recuperation (ISVR). (**d**) Integral squared velocity propulsion (ISVP). (**e**) Average Velocity (AV). Panels (a–d) are displayed normed over distance, and panels (c,d) are transformed into linear space for better interpretability.

Given the synergy between input map-related information with the output DPMADs, the integration of the chosen regression method can take different forms. On one hand, an approach is to build the concept upon one singular model. In this case, only one regressor is trained for the entire prediction task, including all five bin-like-structured DPMADs. On the other hand, a second approach is to train multiple regressors for each of the DPMADs' bin. Because different bins of the DPMADs need to be modelled separately, the second approach was chosen. Furthermore, besides keeping a modularised structure of the solution, this strategy avoids the complexity of one singular model.

The intermediate results show important data insights, namely, that the number of outliers in the peripherical bins negatively affect the prediction and, consequently, the following phases of the pipeline. The outcome indicates the need for a potential redesign of the DPMADs structure, which is subject to future work. However, in this work, we address the drawback by applying an outlier analysis.

Thus, for anomaly detection, Seo compares the possible outlier detection and removal methods [27]. According to this study, Z-score is more appropriate in cases where normal distributions are involved. Therefore, Tukey's method, also known as Interquartile Range (IQR), is further used in our framework [28]. IQR represents a measure of statistical spread and is defined by the difference between the lower (Q1) and upper (Q3) quartiles. With regard to Tukey's method, the observations that fall below $Q1 - 1.5 \cdot IQR$ or above $Q3 + 1.5 \cdot IQR$ are labelled as outliers and further disregarded in the process [28].

From the technical point of view, the bins containing a significant number of outliers are excluded from the regression task and modelled by fixed values, namely, 0 and 1, as these bins are located at the head and tile of the cumulative DPMADs.

Table 2. Default parameters of the linear regression and decision tree algorithms [29].

Linear Regression	**Decision Tree**

fit-intercept = True
normalise = False
n-jobs = 1
	criterion = "gini"
	splitter = "best"
	max-depth = None
	min-samples-split = 2
	min-samples-leaf = 1
	min-weight-fraction-leaf = 0.0
	max-features = None
	random-state = None
	max-leaf-nodes = None
	min-impurity-decrease = 0.0
	min-impurity-split = None
	class-weight = None
	presort = False

2.5. *Application of Classification*

Although Linear Regression provides us with continuous values, representing the DPMADs directly, a Decision Tree is used to return nominal classes, representing the corresponding clusters, from the given input feature vector of representative sets of DPMADs gathered by a clustering done in previous work [30]. It does this by building a tree-like structure, which, starting from its root node, splits the input set with a series of boolean tests. There exist various algorithms for building such data structure, but fundamentally, each new branch of the tree is created to maximise information gain by splitting the input data into unmixed groups. This process yields a nonparametric model, which does not depend on the distribution assumption, that is often prevalent in other approaches. This method is extensively used in literature mainly for its white-box model character, allowing for easy interpretability and keeping a low time complexity. Moreover, it can handle highly-dimensional and non-numerical input data while also providing an importance score of each input feature. Finally, given the simplicity and portability of Decision Trees, they may also be implemented in hardware for fast executions in constrained embedded scenarios. However, the main drawbacks of this method are its tendency for overfitting and the fact that the resulting trees are unstable, so that even small variations in the training data may create completely different resulting models. The Decision Tree algorithm is used as a classifier with the parameters listed in Table 2.

Different from the approach presented in this paper, previous experiments for data reduction with various alternative clustering algorithms and distance metrics have been conducted [30]. The classification part implemented as a second module of the supervised ML solution of the described framework is built upon this work, which provides representative sets of DPMADs with cluster numbers that can be predicted by the above described Decision Tree as depicted in Figure 2.

2.6. *Evaluation Approach*

This section aims to create an in-depth view of the evaluation perspectives, including a deep dive into the fleet and map data. The evaluation examines two main perspectives: the Machine Learning Perspective (MLP) and the Energetic Perspective (EP). Initially, as the solution builds upon ML methods, the system's performance is evaluated from MLP .

For the regression strategy, the bins of the DPMADs are subject to IQR filtering, resulting in a more compact representation. Contrary, the classification strategy maintains the predefined bin-like structure of the DPMADs. The outcomes of both strategies are discussed in Section 3. The same section underlines the need to analyse an ML problem also from a domain-specific point of view.

Additionally, we distinguish the experimental setup by introducing the problem scenarios of a Lost Backend Connection Scenario (LBCS) and Missing Data Scenario (MDS). The LBCS imitates a possible loss of connection between the vehicle and the backend. Whereas, the MDS imitates the absence of some data within the fleet data.

For each of the described problem scenarios, we used two different testing strategies, by considering data from two distinct regions. Specifically, this work uses fleet data recorded in Germany originating from two regions specified by so-called geohash of level 3, which is the output of a public domain geocoding procedure [31]. In more details, a geohash encodes geographic locations into hashes containing digits and letters, so that all geohashes with the same leading identity span a neighbouring geographical area. Specifically, for training and validation of the proposed regression and classification models, data from u28 (Munich area) and u30 (Leipzig area) geohashes are used, as depicted in Figure 4a. Thus, 95% of u28 region's data serve as the training data set, whereas the rest of the u28 and the u30 data are used as the validation subsets. The validation data set from u28 serves for the Application Test Scenario (ATS). The ATS represents the situation within the application situation, where a model trained on one region also is used for the same region. In contrast, the Cross-Evaluation Test Scenario (CTS) uses the u30 validation data, as this could expose overfitting on the training data of the u28 region. For each combination of the presented problem (LBCS and MDS) and test scenarios (CTS and ATS) we performed both classification (Clsf) and regression (Regr) experiments as Table 3 depicts. For Regression, the Linear Regression algorithm, according to Section 2.4, is applied, whereas for the Clsf strategy, the Decision Tree algorithm according to Section 2.5 is used. Both algorithms use default parameter values listed in Table 2, except the max-depth parameter of Decision Tree where the value is set to ten.

Table 3. Evaluation matrix showing the problem scenarios: Lost Backend Connection Scenario (LBCS) and Missing Data Scenario (MDS); test scenarios: Cross-Evaluation Test Scenario (CTS) and Application Test Scenario (ATS); strategies: Regression (Regr) and Classification (Clsf).

Test Scenario	**Test Area**	**Problem Scenario**	
		Lost Backend Connection	**Missing Data**
Cross-evaluation	u30	Regr ↔ Clsf	disregarded
Application	u28	Regr ↔ Clsf	Regr ↔ Clsf

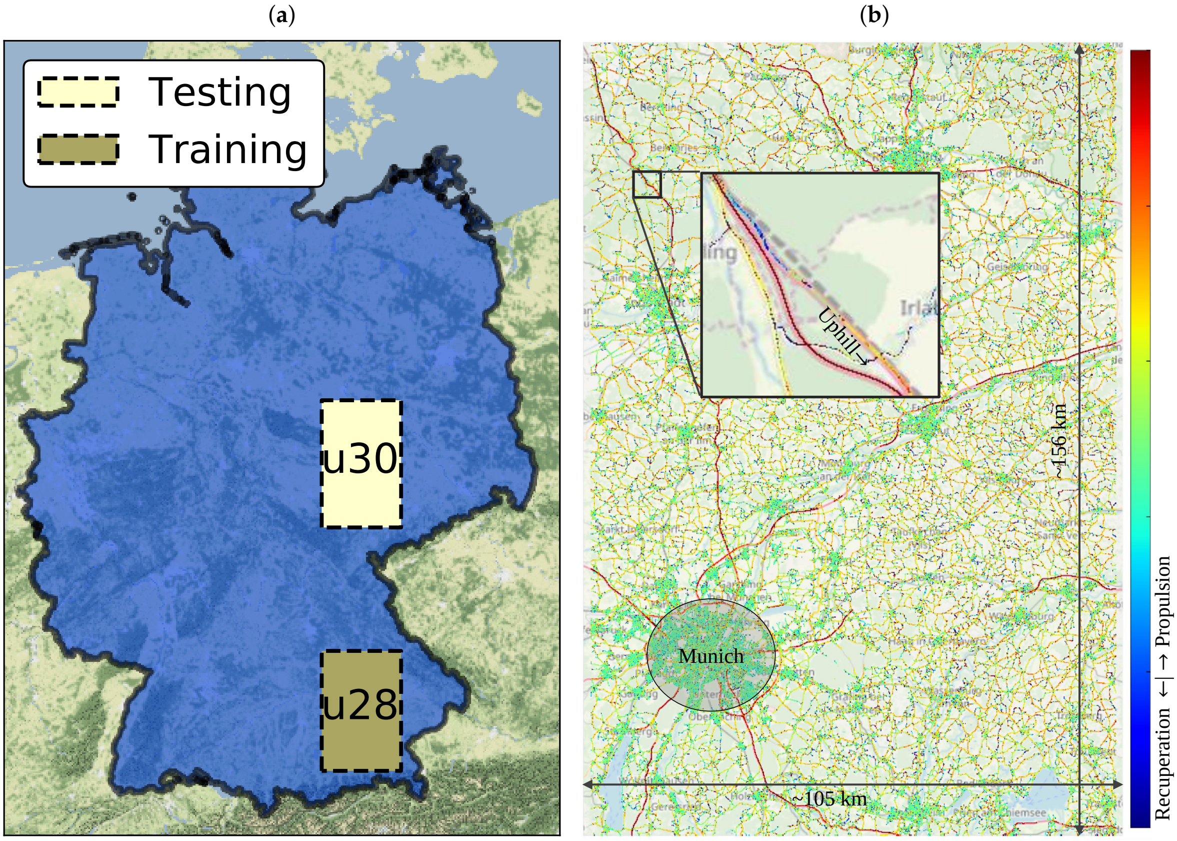

Figure 4. (**a**) Map of the two origin regions, where the data are recorded. The records from the region u28 serve as training data set. The records of u30 serve as cross-validation data set. (**b**) Energetic Map created using the u28 training data, indicating low (blue) to high (red) energy consumption and also energy recuperation (dark blue).

As described previously, LBCS implies that the training phase of the model uses data from u28. However, in this scenario, the model predicts not only within the same geohash, but also a different one, namely, u30. Such a splitting enables the detection of overfitting and ensures an objective evaluation of different strategies. Due to the fact that u30 has no immediate neighbourhood to u28, a fair cross-evaluation environment is ensured by the CTS. Note that the combination of the CTS and the LBCS illustrates the worst-case scenario. In contrast to LBCS, the MDS represents the missing data scenario and involves a data availability check on the backend. Therefore, two scenarios are possible: either the DPMADs are available on the backend, or they need to be predicted by the ML model first. Further, the complete set of DPMADs for each sublink is used to compute energy consumption. For simulating this scenario, the pipeline assumes that the ML model predicts 30% of the data. This simulation set-up imitates the experimentally observed missing data ratio of the u28 validation data set. Overall, the evaluation features three scenarios in different problem and geographical test settings to analyse the two ML approaches. As the upper right cell of Table 3 would depict a weaker worst-case scenario, this work does not address it.

The following amount of data describes the underlying training and testing data sets. The training set contains 95% of available traces from u28, which is 1,491,580 anonymised trace fragments with a total length of 3,503,958 km. As Section 2.1 depicts, each trace is composed of links, which are further divided into sublinks. The sublinks, aligned with both corresponding DPMADs and map features from Section 2.3, embody the input samples of the ML models. Thus, the training set is composed of 291,444 distinct links and 82,757,521 samples. The remaining 5% of traces from u28 are the ATS test set, which is 74,579 trace fragments with a total length of 556,135 km. The LBCS test set consists of the same number of trace fragments, randomly taken from u30 with a full length of 554,366 km.

Based on the data of u28, Figure 4b shows the fleet sourced energetic map summarised for all daytimes. The colours indicate the energy consumption principally for a median driver normed over distance. The unit would be kilowatt hours per 100 km. Dark red refers to very high energy consumption, for example, on the south road going uphill within the zoomed area, whereas dark blue refers to a high recuperation potential, for example, on the north road going downhill within the zoomed area.

With regards to the statistical description of the map-related features, Figures 5 and 6 show the proportion of length for each road class, alongside with the proportion of three speed limit intervals for both validation and training sets. Concerning the weekday feature, in the training data, the proportion of working days is 75% while in the validation sets it is 83% and 88%, within u28 and u30, respectively.

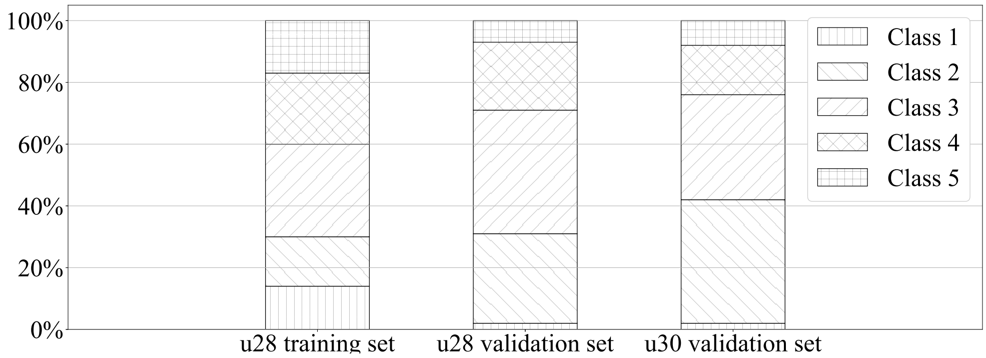

Figure 5. Percentage ratio of the traces to each of the functional road classes in terms of length.

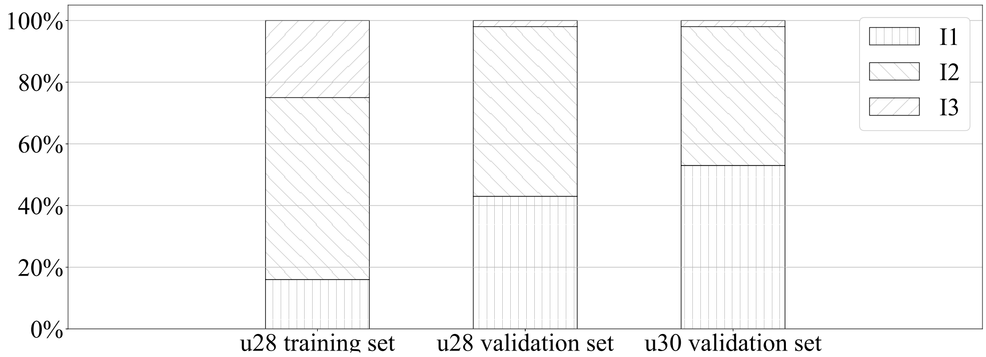

Figure 6. Percentage ratio of the the traces to three intervals of speed limit (km/h): $0 \leq \text{I1} \leq 50$; $50 < \text{I2} \leq 100$; $\text{I3} > 100$.

Prior to the training of the ML models and their evaluation within the specified problem and test scenarios, the handling process of map features need to be addressed. The alignment between the features and the collected DPMADs from the fleet can involve missing map-related information. Thi problem is addressed by the imputation method described below. At link junctions, a preprocessing needs to select the curvature according to the link sequence provided by the map-matching procedure Figure 1. Another curvature-related issue is that some links with a lower traversal and traffic volume have no curvature information included in the map. As they belong to the lowest functional road class category, which presents mainly in residential areas, the imputation assumes straight roads, by imputing null values. Whereas, for sublinks with the missing speed limit, the imputation sets the mean velocity calculated based on the fleet data. Finally, the gaps, in terms of road slope, are filled by averaging the previous and the next sublink's available slope values. To summarise, for the map-related attributes that are not available, the imputation process aims to fill the gap by providing solutions based on neighbouring segments.

3. Results and Discussion

The results obtained by employing the proposed solution are displayed and described in this section. At first, this chapter depicts the results from the MLP followed by the problem-specific EP. Furthermore, paragraphs for the discussion of the outcomes are included.

Specifically, the training used for the training of the ML models hardware containing 2 CPUs, each with ten cores of 2.40 GHz. In terms of memory, the hardware provides 32GB DDR4-RAM. Concerning computational times, the training of the regression models accounts for approximately 39 h, and the training of the classification model takes up to 2 h.

3.1. Machine Learning Perspective

As the described system in this work builds upon supervised ML methods, the results are initially analysed from the MLP described at the beginning of Section 2.6. First, Figure 7a,b shows the performance of both regression and classification models for the LBCS within the CTS that represent the worst-case scenario in Table 3. The lengths difference between the plots from (a) and (b) is the result of the filtering proposed in Section 2.4. To analyse the performances, the mean absolute error (MAE) of each bin's cumulative probability with the DPMADs is used. Accordingly, for the graphs of the acceleration outcomes, there is no significant difference, although the classification method seems to achieve slightly better results. On the contrary, with regards to the velocity outcomes, the regression method presents visibly better results.

Second, Figure 7 shows the performance of both regression and classification models for the LBCS within the ATS. Similar to the previously described set-up, the results from panels (c) and (d) highlight that the classification method presents slightly better outcomes regarding acceleration. Again, the velocity results are better in the case of regression.

Considering the LBCS problem scenario within both CTS and ATS, furthermore, the comparison of the outcomes enables overfitting detection. Accordingly, the comparison of Figure 7a,b versus c,d shows only slight differences, therefore, overfitting seems to be neglectable.

Furthermore, Table 4 summarises the comparison of the graphs for each test scenario in Figure 7 by providing the better performing model for each DPMA and the advantage within the average of all bins' MAE. On one hand, these statistics indicate an equal or better performance of the classification model for the acceleration related DPMA, given both test scenarios. Furthermore, the classification model also performs better for the ISVR, but overall, the greatest advantage is −3.2%. On the other hand, for ISVP and AV, the regression method achieves a magnitude better outcomes with advantages between −10.7% and −28.9%. Regrading ISVR, Figure 7 shows visually favourable plots for the regression, but the statistics indicate it differently. This outcome might be due to the omitted values from the ignored DPMAD bins due to IQR filtering. Thus, the regression model predicts only the values that correspond to the more spreading bins of the DPMADs, whereas the classification includes all bins. Therefore, statistics are not comparable besides giving different indications about which model performs better for the different DPMA. Accordingly, the MLP seems not clear enough, and the need for a more problem-specific perspective arises.

Table 4. Evaluation showing which model performs better in DPMAD prediction separately for each DPMA including deviation in brackets.

Evaluation Region	**u30**		**u28**	
Model	**Recuperation**	**Propulsion**	**Recuperation**	**Propulsion**
Integral acceleration	similar (0.0%)	Clsf (−2.1%)	Clsf (−1.5%)	Clsf (−2.3%)
Integral squared velocity	Clsf (−1.3%)	Regr (−12.4%)	Clsf (−3.2%)	Regr (−10.7%)
Average velocity	Regr (−28.9%)		Regr (−27.2%)	

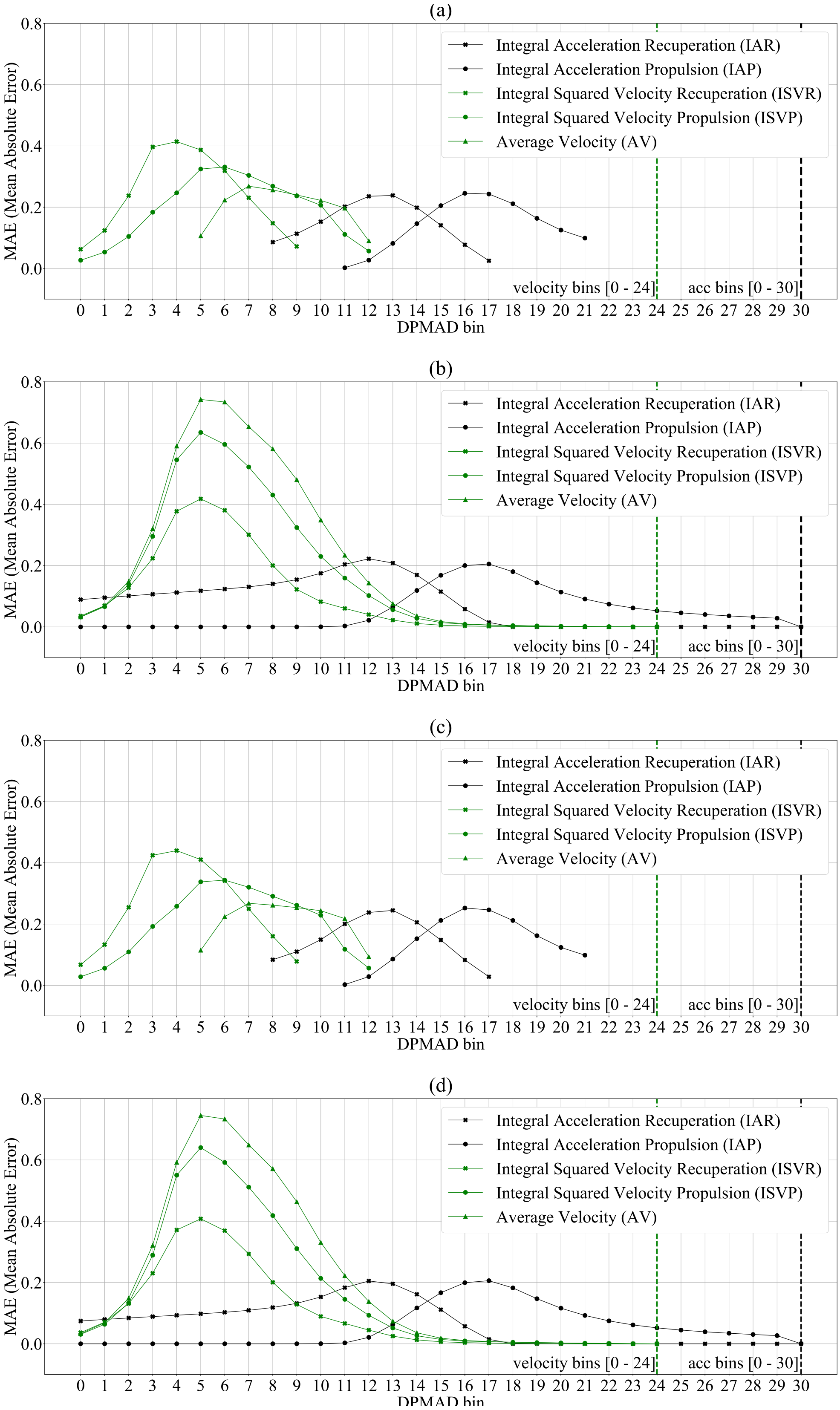

Figure 7. Mean Absolute Errors (MAE) for DPMAD bins using both machine learning (ML) model categories (**a**,**c**) regression and (**b**,**d**) classification. The problem scenario is LBCS, and the test scenario for panels (**a**,**b**) is CTS, whereas for panels (**c**,**d**) is ATC.

The average of all mean absolute errors displayed in each subfigure of Figure 7 should serve as a single overall summary statistic of the MLP on the different models and test scenarios. For both CTS and ATS, the summary statistics of the regression model is equal to 0.19, whereas for the classification model, the values are equal to 0.12. Once again, the classification model's advantage in the second digit of the summary statistics between the two ML methods might occur from the above mentioned bin-wise structure and its application. Overall, the summary statistics show the same values within two digits for the CTS and ATS, which indicates neglectable overfitting. In turn, the overfitting-related observations might be affected by the proximity of the u30 and u28 geographical areas. Another factor might be the season, here spring, as the fleet was recording in April and May. Subsequently, there is the possibility that these overfitting observations could change for other areas or different recording seasons toward worse conclusions, whereas the current results indicate clearly that the set of data is large enough for this analysis.

In conclusion, the MLP indicates the absence of notable overfitting. Furthermore, regression performs in specific velocity-related DPMAs better, whereas, particularly for the acceleration-related DPMAs and velocity-related DPMAs in recuperation, classification presents better outcomes. Subsequently, the classification performs better for most of the DPMAs. However, if regression performs better, the advantage is a lot bigger in comparison to the ones where classification performs superior. Overall, through the analysis conducted from the MLP, the necessity for a problem-specific perspective emerged.

3.2. Energetic Perspective

In contrast to the MLP of looking at the results, Figure 8 shows the problem-specific results in terms of energetic errors within the EP. In regards to the evaluation perspectives described at the beginning of Section 2.6, in the EP only one error value per sample can express the deviation introduced by a model within each evaluation scenario. Figure 8 shows the results for the same scenarios and models like the section before. In panel (a), the figure shows the LBCS tested within the CTS, which is trained on the u28 training set and evaluated on the u30 evaluation set. Here, the regression model introduces a median bias into the ED prediction of −12.6% with a spread of 30.1% in IQR, according to Table 5. Correspondingly, the regression model performs 70.0% better in bias and 80.2% better in IQR spread than the classification one.

Furthermore, panel (b) of Figure 8 shows the result for the ATS based on the evaluation set taken from u28, whereby the problem scenario configurations stay the same. The comparison of the ATS and CTS shows no difference for the classification model and only slight differences in the regression model. In contrast, panel (c) of Figure 8 shows the results for the MDS within the ATS. This setting predicts only a 30% proportion and the rest is assumed to be provided from the fleet data-based DPMADs from the backend. The regression models bias in ED prediction is 7.2% with a spread of 22.6% in IQR, according to Table 5. Once again, the regression performs better than the classification model, which has a 60.0% worse bias and 80.3% higher IQR spread.

Regarding feature-related performance of the ML models, Table 6 depicts the mean energetic error within each road class. Table 6 shows that regression performs best in the case of the road class that has the smallest absolute variation in velocity, which the model needs to learn. By contrast, classification has better outcomes for road classes with a higher level of variance. The result might be due to the fact that both the clustering and classification distinguish more groups of data if there is more spread within those. If there is just a low spread, especially the clustering gives a lower number of representative DPMADs for a data set part of low variance in comparison to a data set part of high variance.

The following paragraphs discuss the beforehand described results to put them into context. Figure 8a shows the test set-up, which represents the overall worst case. It predicts every DPMAD for energetic prediction just based on the trained models in an area of no direct geographic vicinity to the training area, as shown in Figure 4a. In contrast to the indications of the MLP, this worst case, as well as all other test set-ups, show that the regression performs more than two times better than the classification model. Two factors could cause this inferior performance of the classification model. On one hand, the model might serve from the two-step generalisation of the fleet data, as at first, the clustering into representative DPMADs takes place and is followed by the classification model's training which can both introduce errors. On the other hand, this clustering projects the continuous space of the DPMADs provided in step (F) of Figure 1 into a discrete space, which limits the classification model's freedom of action.

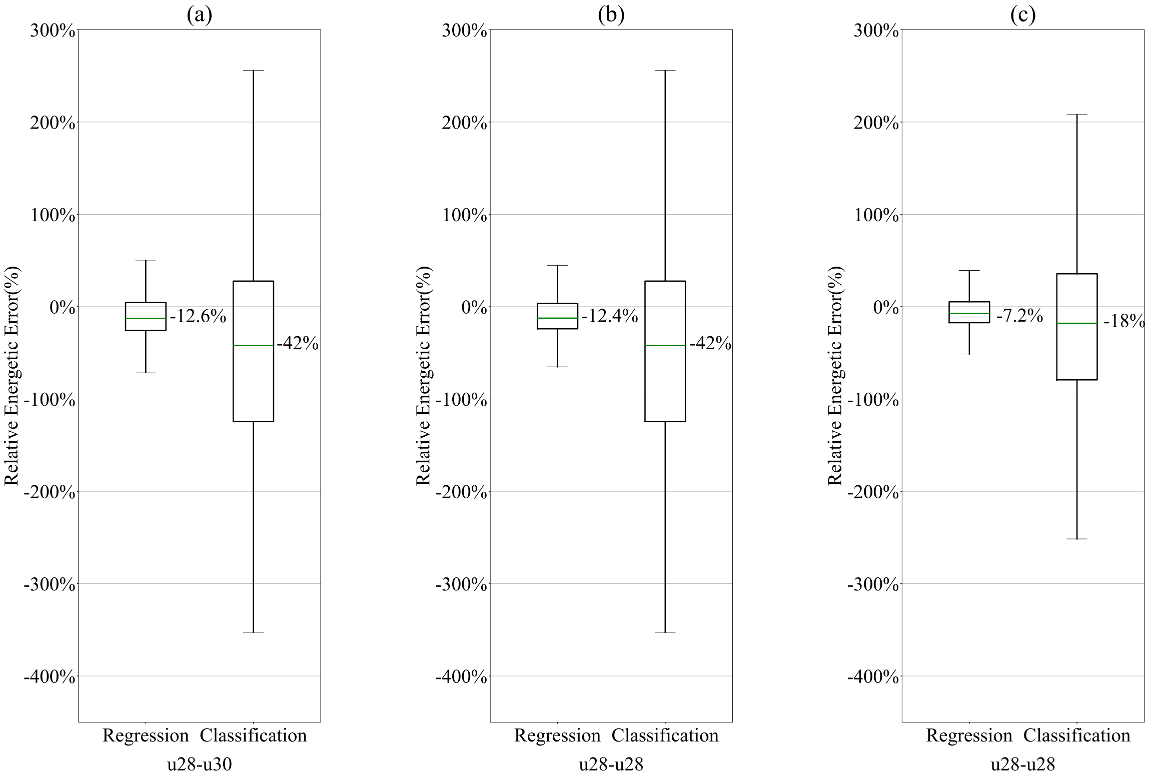

Figure 8. Results displayed considering problem-specific energetic perspective (EP). Therefore, errors shown are percentage errors in predicted ED. Panels (**a**,**b**) show evaluations for the LBCS and a prediction only based on regression and classification models. These evaluations use the model trained on the u28 geohash region and are calculated on the u30 (a) and the u28 (b) region. Panel (**c**) shows results for training and evaluation in u28 region. Accordingly, the regression and classification model predicts only 30% of the segments.

Table 5. Spread in errors for different scenarios from the energetic point of view displaying the interquartile range (IQR) and its borders: The 25%-percentile (Q1) and the 75%-percentile (Q3).

	Lost Backend Connection Scenario				**Missing Data Scenario**	
	u30		**u28**		**u28**	
Statistics	**Regr**	**Clsf**	**Regr**	**Clsf**	**Regr**	**Clsf**
Q1 (%)	−25.6	−124.3	−23.9	−124.3	−17.3	−79.2
Q3 (%)	4.5	27.6	3.5	27.6	5.3	35.6
IQR (%)	30.1	151.9	27.4	151.9	22.6	114.8

This limitation could further cause the absence of any difference for the classification results between panels (a) and (b) in Figure 8, in contrast to some slight differences between the respective regression model results. Both use different data for training and evaluation. Furthermore, the results of panel (a) use evaluation data from a different region. The evaluation data of panel (b) originate from the same region as for the training, as the application of the model would always be trained on all data available and applied to the origin regions. However, the comparison of the results displayed in Figure 8a,b allows the derivation of statements about the level of the model's overfitting. This comparison shows only some slight difference for the regression and for the classification even no difference, which indicates neglectable overfitting. Therefore, the training data set seems to be of sufficient comprehensiveness and size. Moreover, the level of overfitting is so small that only the continuous regression model shows a difference, and the classification model's discrete resolution cannot represent any difference. The different proportions in road classes Figure 5 of the two compared evaluations data sets further support the value of this overfitting evaluation. Nevertheless, another evaluation questioning overfitting could include a test on data of another continent and from another season, like winter, as the cross-validation evaluation set from u30 originates from the same country and season as the training data set. However, the presented evaluations for the proof of a basic concept can already show that the amount of training data seems to be sufficient, and the regression model outperforms the classification one concerning a backup model for the LBCS.

Table 6 gives a more in-depth insight into the energetic deviations within the LBCS problem and ATC test scenario for the regression and classification models. The regression model performs best for the lower important roads, and therefore lower velocities, as there is a lower absolute variance within the data to capture. In contrast, the classification model performs best for the higher priority road class and thus manly higher velocities. This higher deviation occurs as the clustering, which is the basis for the classification, assigns within k-means algorithm fewer representatives to areas of lower variance, and thus the classification has less possibility to distinguish. In turn, the regression is not subject to this resulting discrete resolution, and thus performs better for the road classes with a lower velocity. Even though the classification performs better for higher road classes, it performs for all road classes worse than the regression in the worst road class 4. Accordingly, the regression model is for all road classes, the favourable one.

Table 6. Performance from the best to the worst ranking of ML models in the EP grouped by functional road class with energetic deviation in brackets for the LBCS problem scenario and the ATC test scenario.

Model	
Regr	**Clsf**
Class 5 (6.8%)	Class 2 (27.4%)
Class 3 (14.8%)	Class 1 (38.3%)
Class 2 (15.0%)	Class 5 (49.1%)
Class 1 (16.5%)	Class 4 (54.5%)
Class 4 (18.4%)	Class 3 (57.5%)

The more relevant scenario in a daily application might not be the total loss of backend connection, but the absence of some DPMADs due to an insufficient number of available fleet data samples within a space and time bucket. Figure 8c depicts the results for the latter scenario. This scenario assumes 30% of the DPMADs missing. As this ratio is based on the given two-month fleet data set, a longer period of fleet data recording and a bigger number of recorded traces can decrease this ratio and thus the introduced error. The results are again for classification not as good as for regression. The latter is already promising if compared to some energy prediction models' precision, which provides no dedicated backup models for the case of absent inputs. The analyses of Sautermeister et al. provide an overall range uncertainty margin between 12% and 22.6% for different scenarios [32]. Whereby, the power demand prediction introduces 8.3% to 10.8% uncertainty. The mean absolute percentage error (MAPE) in ED prediction of Masikos et al. is 4.0% [33]. Sarrafan et al. also reports up to 8% error in range prediction [34]. Therefore, the herein presented basic concept using the regression model can already achieve an error of the same magnitude as other methods. Overall, the regression model based on simplistic map features can serve as backup-model to overcome the problems addressed by this work, namely, loss of backend connection and the MDS.

3.3. Comparison of General and Problem-Specific Model Evaluation

This section depicts deductions, from a comparison of the two evaluation perspectives presented in the previous Sections 3.1 and 3.2. The results of the general machine learning perspective indicate that the classification model outperforms the regression one, according to the summary statistics and Table 4. However, for further improvement of the interpretability of these indications, we also performed domain-specific evaluations in Section 3.2, as is usually recommended. This evaluation depicts the energetic error introduced into the ED prediction by the two model options in different scenarios. Whereby this evaluation shows that the regression outperforms the classification model significantly, mostly due to the fact that the classification model serves from errors in the two nested model stages of clustering and classification.

3.4. Outlook and Potential Next Steps

As mentioned in Section 2.2, this work proofs the concept of utilising supervised ML methods to solve the problem of missing data within the energetic map for the EPF. The main contribution of the work lays in the conceptual architecture, which was validated by the results from the problem-specific EP discussed in Section 3.2. In addition to the missing energetic map data issue, the solution can be used as a replacement of the current setup by an onboard deployment. More precisely, the current framework relies on the connection between the vehicle and the backend. However, based on the proposed solution, this could be replaced by the model, which is autonomously deployed in the vehicle.

From scalability and optimisation point of view, the current structure of the concept can be adjusted in its configuration. Concerning features, the set of map attributes can be considerably extended. Specifically, such real-time features like weather or traffic information can be incorporated into the model. Regarding the feature preprocessing, the proof of concept model comprises only single features in a first-order polynomial representation. However, higher polynomial representations can be investigated. As Section 2.3 depicts, certain features allow us to model the influences on both velocity and acceleration in the DPMA. For instance, the combination of the speed limit and functional road class could be beneficial to model different road types' influences on the acceleration DPMAs in a more reliable way. Furthermore, polynomial features of higher orders increase the level of complexity, which can be reproduced by the model. However, all those options for additional features could lead to an explosion of the model and thus, high computational cost.

Regarding the supervised ML methods that take those features as input, both methods used are only a single representative of the available classification and regression algorithms, which are rather simple and come with low computational complexity. Regarding regression, such alternatives as Logistic Regression, Stochastic Gradient Descent or Support Vector Machines can be used as well. Concerning classification, a potential optimisation opportunity could be the usage of such algorithms like Random Forest or shallow as well as deep Neuronal Network. Based on the individual analysis of DPMAD bins, addressed in Section 2.4, future work might even consider different models for each bin as it was done for the regression. Furthermore, the clustering stage included in the pipeline might also be subject to optimisation. The current concept enhances the robustness of the clustering model by considering only DPMADs based on at least three measurements. This value, as discussed in Section 2.5, was chosen based on the idea of having at least a distribution. However, robustness can be further enriched by considering even more measurements.

With regards to the data set, a higher informational value can always be added by bigger data sets. As it was mentioned in Section 3.1, there is still some overfitting as both the training and testing data originate from Germany. Besides, the season in which the fleet data was recorded might also have an impact. Accordingly, data from different seasons can be used in future work. Consequently, an even bigger dataset could cover additional roads and driving circumstances.

Regarding the structure of DPMADs, the current work uses an outlier detection method to determine the bins without substantial variance. However, the peripherical bins can be handled differently, by an adaptation of the bin-wise structure of the DPMADs, also suggested in Section 2.4.

The selection of the DPMAs within the DPMADs for ED prediction could be another opportunity for improvement. The ED calculation based on DPMADs, specified in Section 2.1, currently uses the 50th percentile as the representative for the average driver. This percentile choice is considered as a parameter, which can be optimized by a driver-specific model.

In Section 2.6, several imputation methods for missing features are mentioned. Alternatively, more sophisticated interpolation methods like spline interpolation can be used instead of a next-value interpolation used in this work.

To summarise the outlook, the mentioned modifications offer room for other evaluations in future work on the developed pipeline.

4. Conclusions

To foster the acceptance of emerging electric individual mobility, electric vehicle-specific services can mitigate the user’s concerns regarding limited range and charging-related issues. Therefore, those services should reliably provide range prediction and electric vehicle-specific routing. Both of these services require a reliable route-specific energy demand prediction as an essential component. There are direct, ML-based and analytical energy demand prediction models in the literature. These models source their inputs for prediction from map data or vehicle data. The latter can originate either solely from the prediction vehicle or a fleet of vehicles. If a fleet of vehicles provides the data, routes not driven yet can be predicted as well. Additionally, the fleet data incorporate location-specific influences on energy demand which constitutes an energetic map. Therefore, the objective of the energy prediction framework should be to provide a physically traceable approach that accounts for influences on energy demand, specific for any particular location and time. Therefore, this framework relies on a traceable analytical model based on georeferenced fleet-vehicle data. However, such an approach is limited if data are not available on some roads or time intervals. For further enhancement of the reliability, this paper suggests supervised machine learning-based methods, which generalise the knowledge extracted from available data. In particular, these methods can serve as a backup in two scenarios. On one hand, for the imputation of missing energetic map data, on the other hand, if the connection between vehicle and server is lost, for a complete replacement of the backend’s function to supply energetic map data. A backup for such scenarios aims to enhance the reliability of the proposed energy demand prediction.

This paper aims to be a proof a concept of such a backup for an energy demand prediction, which in turn will enhance its overall performance. Therefore, such classification and regression models as decision tree and linear regression are selected. More sophisticated machine learning models will even further improve the obtained results. Such models could be shallow as well as deep Neuronal Networks or ensemble algorithms that operate with advanced features such as real-time traffic information or weather parameters. However, these approaches are much more computationally demanding and tend to be less scalable. Scalability is particularly essential due to a rather high amount of processed data. Indeed, the training of the models utilises more than 3.5 million kilometres of aggregated fleet data in order to compute five energetic features. The models’ evaluations make use of a cross-evaluation and an application test dataset of another 1.1 million kilometres from two regions in Germany.

The framework was tested from two perspectives, namely, a general machine learning and a problem-specific energetic one. To conclude, the regression model performs better for the addressed problem. Specifically, the regression model introduces 12.6% of energetic error while tested under the worst-case cross-evaluation scenario. This scenario corresponds to a loss of backend connection. This regression model outperforms the classification-based one by 70.0%. Furthermore, in the case of missing energetic map data, the better performing regression model introduces 7.2% bias. This deviation is comparable to the general energy demand prediction errors found in the literature. Therefore, the proof of concept framework presented in this paper shows promising results. Therefore, this machine learning-based fleet data imputation for energy demand prediction can help to increase reliability.

Future work will include a driver-specific energy demand prediction based on this paper's outcomes to improve precision and enhance reliability. Such an energy demand prediction can be used beneficially in multiple ways. As mentioned before, the BEV-specific service of an enhanced range prediction can foster the acceptance of BEVs. Moreover, a BEV-specific trip and charging-stop planning can also contribute. Besides these BEV-specific services, the energetic map could be the basis for the optimisation of charging networks. Furthermore, anticipatory energy management planning could use the energy demand prediction's information to improve efficiency also for different powertrain topologies like a hybrid one.

Author Contributions: Conceptualisation, T.S., M.F. and F.G.; Data curation, T.S.; Formal analysis, M.N.; Methodology, T.S., M.N. and M.S.; Project administration, T.S.; Software, M.N.; Supervision, T.S., M.S., L.T., M.F. and F.G.; Validation, M.N.; Visualisation, T.S., M.N. and L.T.; Writing—original draft, T.S., M.N., M.S. and L.T.; Writing—review & editing, T.S., M.N., M.S., L.T., M.F. and F.G. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by BMW AG within a cooperative PhD-Project with Karlsruhe Institute of Technology, Institute of Vehicle System Technology.

Acknowledgments: We acknowledge support by the KIT-Publication Fund of the Karlsruhe Institute of Technology.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

ATS	Application test scenario
AV	Average Velocity
BEV	Battery electric vehicle
Clsf	Classification
CTS	Cross-evaluation test scenario
DPMA	Driving profile map attributes
DPMAD	Driving profile map attribute distributions
ED	Energy demand
EP	Energetic perspective
EPF	Energy Prediction Framework
GNSS	global navigation satellite system
HVB	High voltage battery
IAP	Integral acceleration propulsion
IAR	Integral acceleration recuperation
IQR	Inter Quartile Range
ISVP	Integral squared velocity propulsion
ISVR	Integral squared velocity recuperation
LBCS	Lost backend connection scenario
MAE	Mean absolute error
MAPE	Mean absolute percentage error
MDS	Missing data scenario
ML	Machine Learning
MLP	Machine Learning perspective
prop	Propulsion
Q1	25%-percentile or lower quantile
Q3	75%-percentile or upper quantile
recu	Recuperation
Regr	Regression

References

1. Yan, Q.; Qin, G.; Zhang, M.; Xiao, B. Research on Real Purchasing Behavior Analysis of Electric Cars in Beijing Based on Structural Equation Modeling and Multinomial Logit Model. *Sustainability* **2019**, *11*, 5870. [CrossRef]
2. Hübner, Y.; Blythe, P.T.; Higgins, C.A.; Hill, G.A.; Neaimeh, M. Use of its to overcome barriers to the introduction of electric vehicles in the North East of England. In Proceedings of the 19th World Congress on Intelligent Transport Systems, Wien, Austria, 22–26 October 2012.
3. Sellschopp, S. Predicting an Energy Consumption of a Vehicle. Patent EP2948357 (A1), 2 December 2015.
4. Lamprecht, A. *Energieprädiktion und Reichweitendarstellung Durch Navigationsdaten im Kraftfahrzeug*; Technische Universität Chemnitz: Chemnitz, Germany, Novermber 2016.
5. Masikos, M.; Theologou, M.; Demestichas, K.; Adamopoulou, E. Machine-learning methodology for energy efficient routing. *IET Intell. Transp. Syst.* **2014**, *8*, 255–265. [CrossRef]
6. Qi, X.; Wu, G.; Boriboonsomsin, K.; Barth, M.J. Data-driven decomposition analysis and estimation of link-level electric vehicle energy consumption under real-world traffic conditions. *Transp. Res. Part D Transp. Environ.* **2018**, *64*, 36–52. [CrossRef]
7. Grubwinkler, S.; Brunner T.; Lienkamp, M. Range Prediction for EVs via Crowd-Sourcing. In Proceedings of the 2014 IEEE Vehicle Power and Propulsion Conference (VPPC), Coimbra, Portugal, 27–30 October 2014. [CrossRef]
8. de Cauwer, C.; Verbeke, W.; van Mierlo, J.; Coosemans, T. A Model for Range Estimation and Energy-Efficient Routing of Electric Vehicles in Real-World Conditions. *IEEE Trans. Intell. Transp. Syst.* **2019**, 1–14. [CrossRef]
9. Yang, J.Y.; Chou, L.D.; Chang, Y.J. Electric-Vehicle Navigation System Based on Power Consumption. *IEEE Trans. Veh. Technol.* **2016**, *65*, 5930–5943. [CrossRef]
10. Wang, J.; Besselink, I.; Nijmeijer, H. Battery electric vehicle energy consumption prediction for a trip based on route information. *Proc. Inst. Mech. Eng. Part D J. Autom. Eng.* **2018**, *232*, 1528–1542. [CrossRef]
11. Schlomer, G.; Bauman, S.; Card, N. Best Practices for Missing Data Management in Counseling Psychology. *J. Couns. Psychol.* **2010**, *57*, 1–10. [CrossRef] [PubMed]
12. Jerez, J.M.; Molina, I.; García-Laencina, P.J.; Alba, E.; Ribelles, N.; Martín, M.; Franco, L. Missing data imputation using statistical and machine learning methods in a real breast cancer problem. *Artif. Intell. Med.* **2010**, *50*, 105–115. [CrossRef] [PubMed]
13. Jeng, R.S.; Kuo, C.Y.; Ho, Y.H.; Lee, M.F.; Tseng, L.W.; Fu, C.L.; Liang, P.F.; Chen, L.J. Missing data handling for meter data management system. In Proceedings of the Fourth International Conference on Future Energy Systems, Berkeley, CA, USA, 22–24 May 2013; Culler, D., Ed.; ACM: New York, NY, USA, 2013; p. 275. [CrossRef]
14. Wynter, L.; Min, W.; Morris, B.G. Method and Structure for Vehicular Traffic Prediction with Link Interactions and Missing Real-Time Data. U.S. Patent 8755991(B2), 11 March 2010.
15. Scheffer, J. Dealing with Missing Data. *Res. Lett. Inf. Math. Sci.* **2002**, *3*,153-160 .
16. Lakshminarayan, K.; Harp, S.A.; Samad, T. Imputation of Missing Data in Industrial Databases. *Appl. Intell.* **1999**, *11*, 259–275. [CrossRef]
17. Laña, I.; Olabarrieta, I.I.; Vélez, M.; Ser, J.D. On the imputation of missing data for road traffic forecasting: New insights and novel techniques. *Transp. Res. Part C Emerg. Technol.* **2018**, *90*, 18–33. [CrossRef]
18. Leke, C.; Twala, B.; Marwala, T. Modeling of missing data prediction: Computational intelligence and optimization algorithms. In Proceedings of the IEEE International Conference on Systems, Man and Cybernetics (SMC), San Diego, CA, USA, 5–8 October 2014; IEEE: Piscataway, NJ, USA, 2014; pp. 1400–1404. [CrossRef]
19. Lu, X.; Si, J.; Pan, L.; Zhao, Y. Imputation of missing data using ensemble algorithms. In Proceedings of the Eighth International Conference on Fuzzy Systems and Knowledge Discovery (FSKD), Shanghai, China, 26–28 July 2011; Ding, Y., Ed.; IEEE: Piscataway, NJ, USA, 2011; pp. 1312–1315. [CrossRef]
20. Lamprecht, A. Verfahren zum Bestimmen eines zu Erwartenden Verbrauchswertes eine Kraftwagens. Patent DE102012008688 (A1), 31 December 2013.
21. Newson, P.; Krumm, J. Hidden Markov map matching through noise and sparseness. In Proceedings of the 17th ACM SIGSPATIAL, Seattle, WA, USA, 4–6 November 2009; pp. 336–343. [CrossRef]

22. Mitchell, T.M. *Machine Learning*; McGraw-Hill Series in Computer Science; McGraw-Hill: New York, NY, USA, 2010.
23. Fan, J.; Wu, L.; Zhang, F.; Cai, H.; Zeng, W.; Wang, X.; Zou, H. Empirical and machine learning models for predicting daily global solar radiation from sunshine duration: A review and case study in China. *Renew. Sustain. Energy Rev.* **2019**, *100*, 186–212. [CrossRef]
24. Seyedzadeh, S.; Rahimian, F.P.; Glesk, I.; Roper, M. Machine learning for estimation of building energy consumption and performance: A review. *Vis. Eng.* **2018**, *6*, 265. [CrossRef]
25. Severson, K.A.; Attia, P.M.; Jin, N.; Perkins, N.; Jiang, B.; Yang, Z.; Chen, M.H.; Aykol, M.; Herring, P.K.; Fraggedakis, D.; et al. Data-driven prediction of battery cycle life before capacity degradation. *Nat. Energy* **2019**, *4*, 383–391. [CrossRef]
26. Fox, J. *Applied Regression Analysis and Generalized Linear Models*; Sage Publications: Thousand Oaks, CA, USA, 2015.
27. Seo, S. A Review and Comparison of Methods for Detecting Outliers in Univariate Data Sets. Master's Thesis, University of Pittsburgh, Pittsburgh, PA, USA, 2006.
28. Tukey, J. *Exploratory Data Analysis*; Addison-Wesely: Boston, MA, USA, 1977.
29. Buitinck, L.; Louppe, G.; Blondel, M.; Pedregosa, F.; Mueller, A.; Grisel, O.; Niculae, V.; Prettenhofer, P.; Gramfort, A.; Grobler, J.; et al. API design for machine learning software: Experiences from the scikit-learn project. In Proceedings of the ECML PKDD Workshop: Languages for Data Mining and Machine Learning, Prague, Czech Republic, 23–27 September 2013; pp. 108–122.
30. Kiener, M. Clustering of Fleet Data for Energy Prediction. Master's Thesis, Technische Universität München, Munich, Germany, 2019.
31. Morton, G.M. *A Computer Oriented Geodetic Data Base and a New Technique in File Sequencing*; International Business Machines Company Co. Ltd.: Ottawa, ON, Canada, 1966.
32. Sautermeister, S.; Falk, M.; Baker, B.; Gauterin, F.; Vaillant, M. Influence of Measurement and Prediction Uncertainties on Range Estimation for Electric Vehicles. *IEEE Trans. Intell. Transp. Syst.* **2017**, *19*, 2615–2626. [CrossRef]
33. Masikos, M.; Demestichas, K.; Adamopoulou, E.; Theologou, M. Mesoscopic forecasting of vehicular consumption using neural networks. *Soft Comput.* **2015**, *19*, 145–156. [CrossRef]
34. Sarrafan, K.; Muttaqi, K.M.; Sutanto, D.; Town, G.E. A Real-Time Range Indicator for EVs Using Web-Based Environmental Data and Sensorless Estimation of Regenerative Braking Power. *IEEE Trans. Veh. Technol.* **2018**, *67*, 4743–4756. [CrossRef]