Route2Vec: Enabling Efficient Use of Driving Context through Contextualized Route Representations

Philipp Hallgarten
Porsche AG
Stuttgart, Germany
TU Munich
Technical University of Munich, Germany
philipp.hallgarten1@porsche.de

Thomas Kosch
HU Berlin
Berlin, Germany
thomas.kosch@hu-berlin.de

Tobias Grosse-Puppendahl
Porsche AG
Stuttgart, Germany
tobias@grosse-puppendahl.com

Enkelejda Kasneci
Human-Centered Technologies for Learning
Technical University of Munich
Munich, Germany
enkelejda.kasneci@tum.de

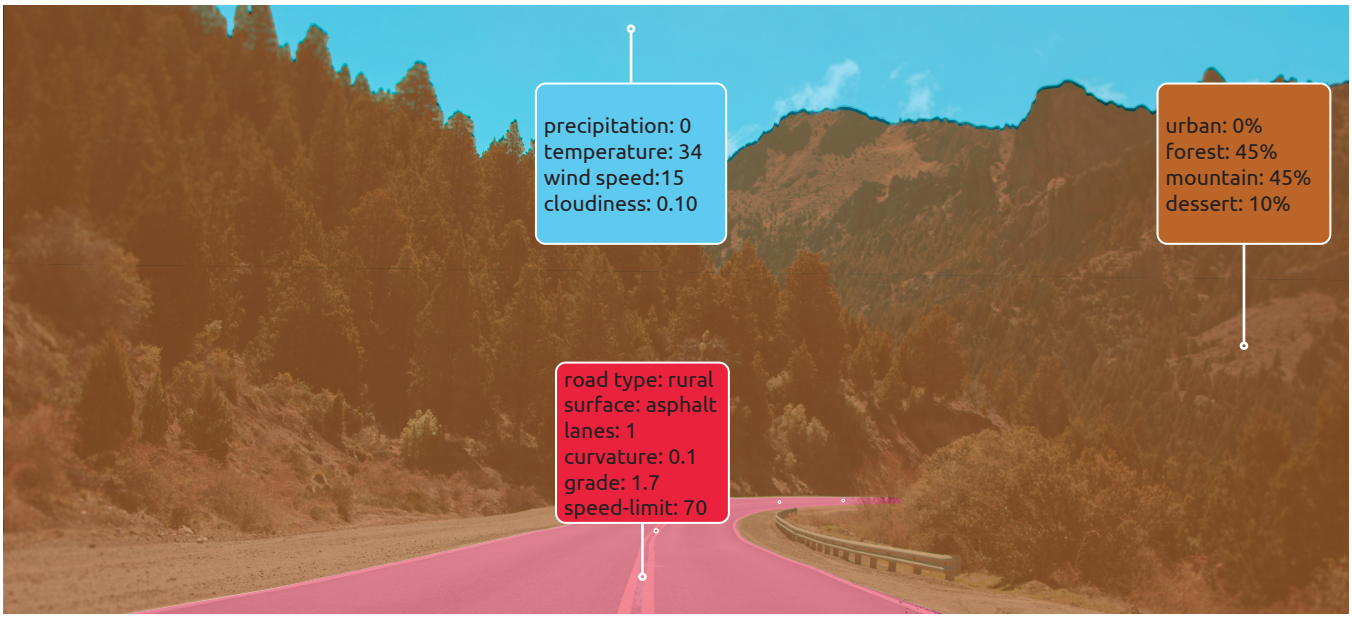


Figure 1: We present *Route2Vec*, a system that encodes variable-length sequences of route context, such as road type, traffic, and weather, into fixed-size semantic embeddings. The learned embeddings preserve key route-specific features, and similar sets of route context are encoded to similar embeddings. This allows for efficient comparison of route context across different scenarios using simple metrics like Euclidean distance. Thus, *Route2Vec* builds the basis for enabling context-aware interactions in mobile platforms such as cars.

Abstract

Understanding how vehicle occupants experience their journey is key to designing adaptive in-car systems. The environments they encounter, ranging from road types and traffic patterns to weather conditions, shape their mental and emotional states during a ride. Yet, leveraging this contextual information remains a challenge due to its heterogeneous nature, comprising diverse data types, such as categorical, numerical, and boolean values of various scales. We introduce *Route2Vec*, an attention-based framework that encodes variable-length sequences of route context into compact, semantically meaningful embeddings using a self-supervised learning pipeline. These fixed-size representations allow for efficient comparisons between different driving situations using common similarity metrics such as Euclidean distance. Through linear probing and

MuC '25, Chemnitz, Germany

ACM ISBN 979-8-4007-1582-2/25/08
https://doi.org/10.1145/3743049.3743056

qualitative analysis of the embedding space, we show that *Route2Vec* reliably captures salient, route-specific characteristics. *Route2Vec* simplifies context-aware in-vehicle interaction by enabling designers to rapidly prototype intelligent in-vehicle interfaces. We make our trained models and code[1] publicly available to foster research in this area.

CCS Concepts

• **Information systems**; • **Human-centered computing** → **Human computer interaction (HCI)**; • **Computing methodologies** → Learning latent representations;

Keywords

Context-Aware Systems, Representation Learning

ACM Reference Format:
Philipp Hallgarten, Thomas Kosch, Tobias Grosse-Puppendahl, and Enkelejda Kasneci. 2025. Route2Vec: Enabling Efficient Use of Driving Context through Contextualized Route Representations. In *Mensch und Computer 2025 (MuC '25), August 31–September 03, 2025, Chemnitz, Germany*. ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/3743049.3743056

1 Introduction

Vehicles have evolved from simple transportation tools into sophisticated platforms that integrate advanced technologies to deliver rich and personalized user experiences [27]. Consequently, they offer an unprecedented range of functionalities, enabling drivers and passengers to adjust settings such as lighting, scent, curated soundscapes, and entertainment with just a few taps. At the same time, prior studies have shown that contextual factors that drivers and passengers are exposed to while driving, including road types, traffic density, and weather conditions, are largely influential on drivers' experiences during the ride. Among other, they correlate with drivers' emotional states [5], influence their interruptibility [28], or determine their route preferences [3]. For example, driving along a highway segment during heavy rain at night entails a vastly different context than a city road on a sunny afternoon, requiring systems to adapt accordingly. We refer to these variables as *driving context* in this work. As vehicles become more adaptive, leveraging this driving context for personalization is a promising direction. However, doing so introduces significant technical challenges. Contextual data is inherently heterogeneous, comprising categorical, numerical, and boolean values of diverse scales and distributions. Thus, traditional similarity metrics, such as Euclidean distance, are ill-suited for handling this mixed-type data. Additionally, context often evolves over time, requiring systems to account for data sequences rather than isolated snapshots. Thus, comparing two sets of context requires capturing the temporal and relational structure within the data.

We address this challenge by introducing *Route2Vec*, an attention-based framework for encoding variable-length sequences of context variables into fixed-size semantic embeddings. By encoding them into a compact representation, *Route2Vec* enables straightforward comparisons of routes using common metrics such as Euclidean distance. It is trained in a self-supervised learning pipeline on a synthetic dataset, omitting the need for costly manual labeling. Through quantitative and qualitative analyses, we evaluate four versions of *Route2Vec*– small, medium, large, and extra-large. A linear probing analysis demonstrates that the learned embeddings capture semantically meaningful information about routes, such as road type distributions or speed profiles. Finally, a qualitative exploration of the embedding space confirms that sequences with similar sets of contextual characteristics are encoded in close proximity to the learned embedding space, thus validating the framework's capacity to preserve contextual similarities.

We summarize the key contributions as follows:

- **C1:** We introduce *Route2Vec*, a novel attention-based system for encoding sequences of road context into semantic embeddings,
- **C2:** we present a synthetic dataset for training *Route2Vec* in a self-supervised pipeline, and
- **C3:** we present comprehensive analyses of the learned embedding space learned by *Route2Vec*.

Thus, *Route2Vec* enables the design of context-aware vehicle interfaces that go beyond static personalization by dynamically adapting to the driver's situational needs. By understanding and comparing routes based on their contextual characteristics, in-vehicle systems can reduce interaction effort, ultimately enabling more seamless and supportive user experiences.

2 Related Work

Route2Vec's core underlying idea of encoding route context into context-aware embeddings is motivated by previous research on the significance of driving context for humans' internal states and, thus, for user-facing mobile applications. Methodologically, it draws from research on self-supervised embedding learning techniques. We discuss both areas of research and their relevance for *Route2Vec* in this section.

2.1 Significance of Driving Context for Mobile Applications

The prevalent influence of driving context, including information on weather, traffic, road properties, and daytime, on drivers' emotions [3–6, 18, 21], drivers' interruptibility [15, 28], and drivers' internal states [10, 11, 29, 30] has been thoroughly demonstrated in previous work. For example, Kim et al. presented a dataset and neural network-based system that can predict drivers' interruptibility from driving context [15]. Bethge et al. built a virtual emotion sensor for drivers based on vehicle- and traffic dynamics, road characteristics, weather information, and in-vehicle context [5]. In subsequent work, Bethge et al. present *HappyRouteing*, a system that leverages this correlation between driving context and driver-felt emotions to compute an emotional map layer that allows users to route along the happiest path between two points [3]. Rung et al. take a comparable approach by proposing *Autobahn*, a system for generating scenic routes. However, in contrast to *Happy Routing*, *Autobahn* is based on the visual characteristics of route segments extracted from street view images [25]. Exploring the influence of road context for in-vehicle interactions, Wolf et al. present HMInference, a machine learning system that predicts future interaction modalities for automotive UIs based on users' interaction history

[1] https://github.com/philipph77/Route2Vec/

and driving context [35]. Finally, Wiedner et al. propose an intelligent User Interface for car infotainment systems that recommends interactable UI items based on driving context [34].

The above-mentioned work shows the versatility and widespread applicability of driving context in mobile platforms. However, these systems use driving context to predict a task-specific label or outcome. While effective for specific applications, they do not provide a means to explicitly compare different sets of driving context. Such comparisons are essential for three reasons. First, encoding driving context into reusable representations allows systems to generalize across multiple tasks, reducing required dataset size for model training and omitting the need for feature engineering. Second, enabling explicit comparisons increases the interpretability of such systems as it allows us to explain why similar labels are predicted for different scenarios by identifying similarities in their underlying contexts. Third, efficient comparison of driving contexts provides a foundation for broader, scalable, context-aware systems that operate in diverse environments. *Route2Vec* addresses these challenges by encoding context into compact, semantic embeddings, enabling efficient and interpretable comparisons across contexts.

2.2 Self-Supervised Embedding Learning

To encode variable-length sequences of context variables into semantic embeddings, we draw from research on self-supervised representation learning. Embedding learning is a well-known challenge in natural language processing (NLP). The authors of [22] present the renowned *word2vec* architecture, trained on a large-scale dataset ($1.6B$ samples) to encode embeddings for words so that words with similar meaning are encoded to similar embeddings. The success of this technique has inspired its application beyond NLP, with adaptations for time-series data, spatial data, and multimodal inputs. Hallgarten et al. propose a mechanism combining momentum contrast with a reconstruction task to learn embeddings for EEG time-series and body-worn IMU sensors [12]. In [19], the authors propose *Space2Vec*, a representation learning model for spatial data that encodes absolute positions and spatial relationships. For autonomous driving applications, Malawade et al. introduce *Roadscene2Vec*, a tool that can extract, among other things, the position of other vehicles around the ego vehicle and embed them into a road scene-graph [20]. Finally, Baevski et al. introduce *data2vec*, a generalizable self-supervised learning mechanism based on momentum contrast to learn an embedding function for images, text, and audio data [1].

While these projects served as inspiration for *Route2Vec*, they differ in three ways. First, *Route2Vec* learns to project routes into an embedding space that preserves road context information. Second, *Route2Vec* is designed to work with potentially small ($60k$ samples) and unlabeled datasets. Third, *Route2Vec* allows the retrieval of routes with similar sets of context through a nearest neighbor search in the embedding space.

3 Methodology

Our methodology to learn semantically meaningful driving context embeddings is two-fold. First, we propose a dataset generation procedure, which can be scaled to high sample sizes and is generalizable to most regions on Earth[2]. Next, we present an encoding architecture that projects sequences of road context to meaningful embeddings without relying on labeled data.

3.1 Terminology

Schmidt et al. define context as a situation, and the environment a device or user is in that is identified by a unique name and comprises a set of relevant features [26]. Following this definition, we distinguish different types and sets of context variables in this work. An overview is presented in Figure 2. We cue the set composed of all variables that influence human beings while driving as *driving context*. For this work, we consider the set to be composed of three main clusters. First, *Vehicle Context*, which describes the vehicle's current state, such as its speed or number of occupants. Second, *Passenger Context* describes the state of the human beings in the vehicle. This can be, for example, the driver's emotion or mental state. Third, *Route Context* summarizes all variables related to the road and environment, such as speed limits, greenness of the area, or the road type. This route context can be further split into dynamic variables, such as weather- or traffic-related ones, and static variables. Dynamic context variables vary over time for a given location, e.g., the weather can change from cloudy to rainy. In contrast, static context does not vary over time for a given location or varies so slowly that it can be neglected; e.g., a road's surface can be assumed not to change at a given position in the relevant time interval. We name this set *Road Context*. Road context contains the road type of the street, its surface, or speed limits, as well as the land use around the road, i.e., whether it is an industrial area or a rural area.

We argue that all three types of route context influence the users' perception of the situation. The inclusion of dynamic variables in context embeddings increases their richness; however, it reduces their re-usability, as it is, for example, not practical to store them in lookup tables. In this work, we focused on road context only to make our solution applicable to the widest range of possible scenarios. Encodings of dynamic context could be generated and added using the same methodology.

3.2 Synthetic Dataset Generation Procedure

3.2.1 Notation. We consider the dataset $\mathcal{D}$ as a set of S samples s_i, *i.e.* $\mathcal{D} = \{s_i\}_{i=1}^{S}$. Hereby, each sample s_i is a sequence of context vectors $x_i^{<j>} \in \mathbb{R}^F$, with $j \in [1, L_i]$ being the segment ID, F being the number of context features, and L_i being the length of the segment s_i. Note that, without loss of generality, this notation implies that the sequences are composed of segments of *equal context i.e.*, segments where the context vector is constant but not of segments of equal length or travel time.

3.2.2 Road Graph Generation. Our dataset is produced by a scalable data generation procedure. It starts by defining a bounding box within which the sequences shall be located. We use the *OSMnx* [7] package to load a geospatial OpenStreetMap Graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ consisting of road segments $\mathcal{E}$, and intersections $\mathcal{V}$ for the defined bounding box, both annotated with road context features. An

[2] coverage through map services is required

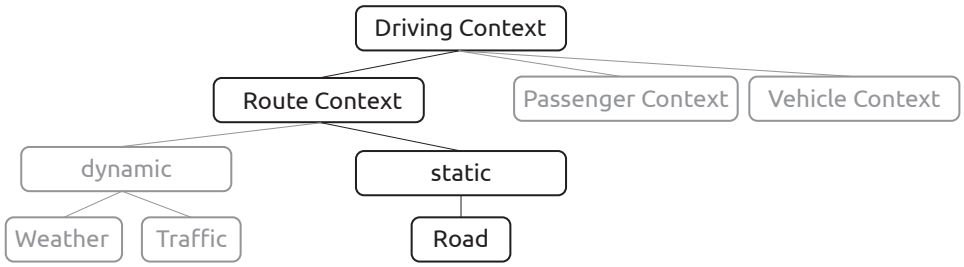
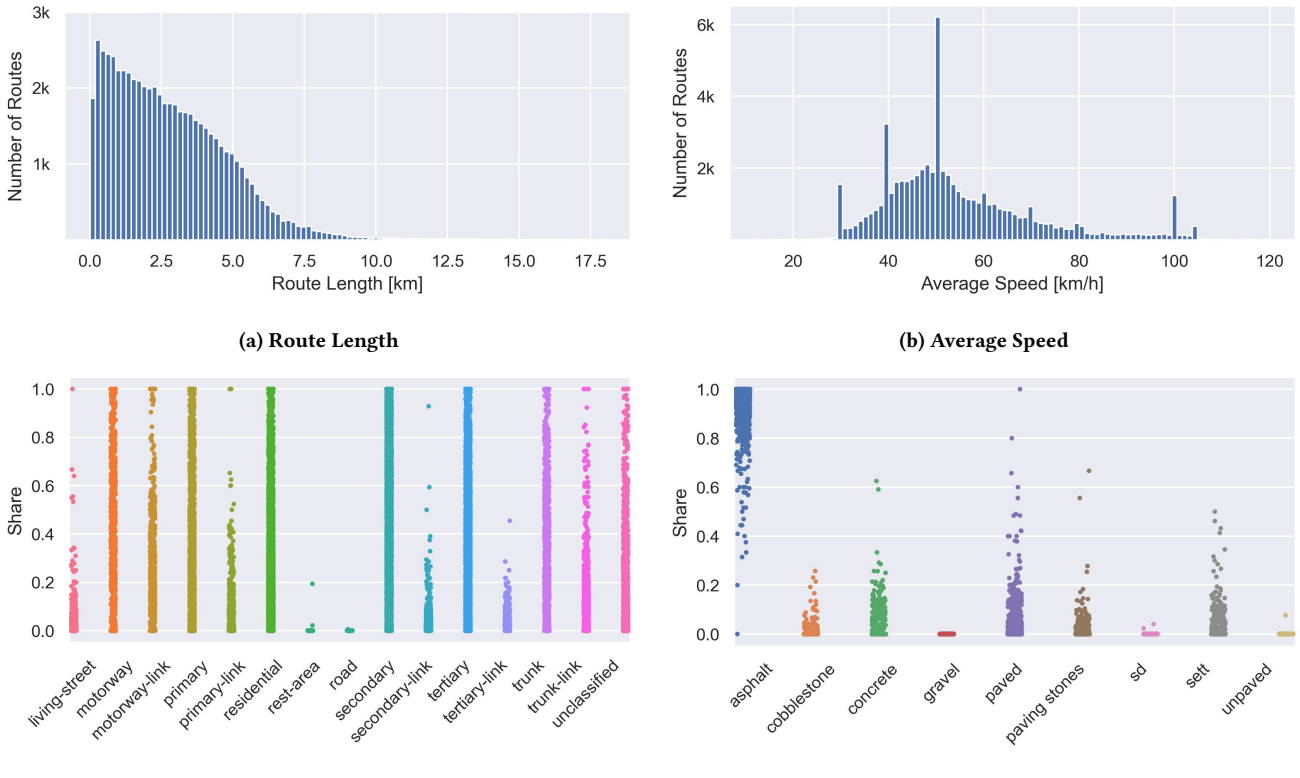


Figure 2: We define *Driving Context* as the set of context variables that influence human beings while driving. It is composed of three main clusters: *Route Context*, *Passenger Context*, and *Vehicle Context*. Route context can be further distinguished into dynamic variables (weather- or traffic-related) and static variables (*Road Context*).



(a) Route Length

(b) Average Speed

(c) Road Type Shares

(d) Surface Shares

Figure 3: Shown are characteristics of the sequences of road context we sample from a contextual street graph to create a synthetic dataset. For the exemplary features of the routes in this dataset, such as (a) the route lengths, (b) the average speeds, (c) the road type shares, and (d) the surface shares, we observe long-tailed distributions, hence the sampled routes cover a wide variety of contexts. For (c) and (d), 10000 routes of the dataset were randomly sampled to avoid visual clutter.

overview of the features provided through OpenStreetMap is presented in Table 1. Further, we annotate the nodes $\mathcal{V}$ with elevation data provided by the Copernicus Land Monitoring Service [8] and thereafter calculate the grade for the edges $\mathcal{E}$ of our graph. The context vectors of the edges form a mixture of categorical data, boolean data, and numerical data.

3.2.3 Route Sampling Procedure. We aim for the samples in our dataset to represent a range of contexts that a person might encounter outdoors, for example, when driving a car. We argue that using context vectors of one single position would not be meaningful, but rather sequences of context vectors. We will name these sequences *routes* in the following. To obtain sequences of road context from the road graph $\mathcal{G}$, we randomly sample $2N_S$ uniform

Table 1: Features provided through OpenStreetMap [23]. The features below the horizontal line were used to learn the embedding function.

Feature	Type	Used for Training	Example	Description
lat	float	✗	38.775845	latitude of the start of the segment (EPSG Projection)
lng	float	✗	8.182932	longitude of the start of the segment (EPSG Projection)
x	integer	✗	408103.7	X of the start of the segment (UTM Projection)
y	integer	✗	4418515.0	Y of the start of the segment (UTM Projection)
width	float	✗	10.2 m	width of the segment
bearing	float	✓	57.50	bearing of the segment
bridge	boolean	✓	False	segment is a bridge
curvature	float	✓	0.027909	the Menger curvature [17]
grade	float	✓	-0.015	grade of the segment
junction	boolean	✓	True	segment is a junction
lanes	integer	✓	3	lanes of the segment
length	float	✓	31.743 m	length of the segment
oneway	boolean	✓	True	segment is a one-way road
road type	categorical	✓	residential	road type of the segment
speed	integer	✓	50 km/h	speed on the segment
surface	categorical	✓	asphalt	surface of the segment
travel time	integer	✓	2.3 s	time to pass the segment
tunnel	boolean	✓	False	segment is a tunnel

graph-constrained points, of which half are referred to as the start point and the other half as the destination point. Next, each start point is randomly matched to a destination point without replacement, and the shortest path between these pairs is calculated. By assuring that each point is used only once as a start or destination point in our dataset, we reduce the likelihood of edges near these points being over-represented in the sampled paths. Each path is represented as a sequence of graph edges that one has to travel along to get from the start point to the destination point. By concatenating the context vectors of the edges along a path to a sequence, we obtain a sample s_i for our dataset. A detailed overview of the distributions of some exemplary features is shown in Figure 3. We observe long-tailed distributions, e.g., over the average speed of the routes; hence, the sampled routes cover a wide variety of contexts.

3.2.4 Preprocessing. Our final dataset consists of approximately 60 000 routes, of which we use 60% as training data, 20% as validation data, and the remaining 20% as testing data. Numerical features are standardized; categorical features are one-hot encoded. As we find the data coverage for the feature *width* to be rather low, we refrain from using it in our study. Missing values further present in the features *curvature*, *lanes*, and *surface* are filled with zeros.

3.3 Route2Vec - Road Context Embedding Framework

3.3.1 Learning Pipeline. Our proposed road-context embedding framework, *Route2Vec*, trains a deep neural encoding architecture in a custom self-supervised representation learning pipeline. An overview of the learning pipeline is presented in Figure 4. The goal of the training is to make the encoder learn a meaningful projection function from sequences of context s_i to embeddings e_i. To these means, we define a set of T proxy tasks to obtain a supervision signal that guides the training process. Multiple reconstruction models are trained to recover certain contextual information from the embeddings, for which the labels can be calculated on the fly from the data itself. After training, these reconstruction models can be omitted, and the encoder can extract context-embedding from unlabeled data.

3.3.2 Encoding Architecture. The neural encoding architecture is inspired by *BERT* [9] and consists of a tokenizer, positional embeddings, and a stack of N_T transformer encoders. The model learns a mapping function f parameterized through θ_{ENC} with $f_{\theta_{\text{ENC}}} : \mathbb{R}^{LxF} \rightarrow \mathbb{R}^D$. As in *BERT*, the segments $x_i^{<j>}$ of sample s_i are first mapped to tokens $z_i^{<j>}$ by a tokenizer with parameters $\theta_{\text{Tok}} \subseteq \theta_{\text{ENC}}$. Here, the tokenizer is implemented as a single linear layer model. Next, a trainable token $z_i^{<0>} \subseteq \theta_{\text{ENC}}$ is added to the beginning of the sequence, which we refer to as *CLS* token. It is randomly initialized and learned during training. A positional embedding vector is added to each of the elements in the sequence as introduced in [32]. These vectors are composed of sinusoidals and are unique for every position j in the sequence, allowing the model to exploit positional information. The so obtained tokens are then passed through a transformer-based encoder, parameterized through $\theta_{\text{Trans}} \subseteq \theta_{\text{ENC}}$, outputting a sequence of representations $e_i^{<j>}$, where we use the one at first position $e_i^{<0>}$ as context embedding. For simplicity, we will further denote the route context embedding with e_i.

3.3.3 Classifiers' Architecture. The route context embeddings e_i are used as input for T linear reconstruction models implemented as single dense layer models with parameters $\theta_{\text{REC}^{(t)}}$, $t \in [1, T]$ to retrieve certain context-information $y^{(t)}$ of the input, that can be calculated on-the-fly. Thus,

$$f_{REC^{(t)}} : \mathbb{R}^D \rightarrow \mathbb{R}^{C^{(t)}}, \tag{1}$$

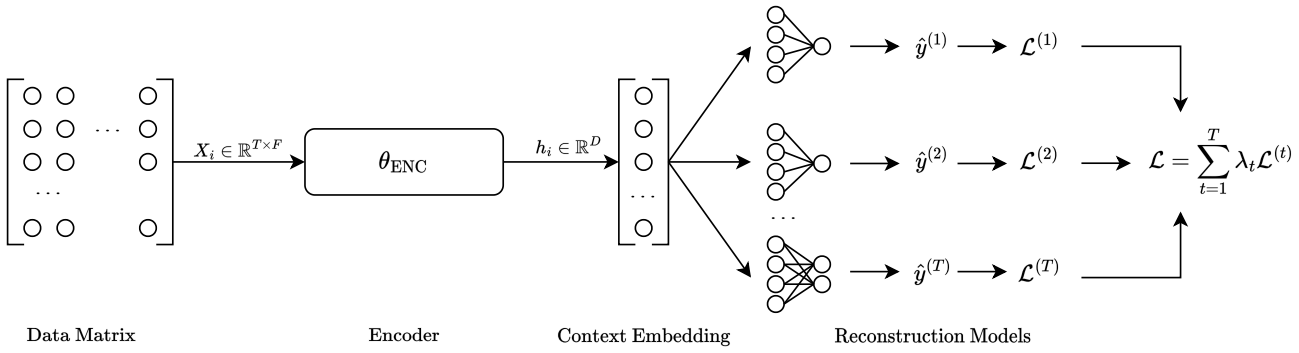


Figure 4: *Route2Vec*'s architecture consists of an encoder and multiple reconstruction models. The encoder projects sequences of road context to semantic embeddings. During self-supervised training, multiple linear reconstruction models try to reconstruct encoded context information from the embeddings, leading the encoder to learn a useful projection function. The labels for the reconstruction models can be calculated on the fly from the data itself.

and

$$\hat{y_i}^{(t)} = \theta_{\text{REC}^{(t)}} e_i. \tag{2}$$

We define the following contextual information of the routes as labels for the reconstruction tasks: (1) *summed travel time*, (2) *summed route length*, (3) *mean curvature*, (4) *mean absolute grade*, (5) *road type share*. For tasks (1) - (4) we encounter ordinary regression task ($C^{(t)} = 1$), thus we use a mean-squared error (MSE) loss $\mathcal{L}^{(t)} = \|y_i^{(t)} - \hat{y_i}^{(t)}\|_2^2$ for training of the reconstruction models' weights. In the case of the task (5), we obtain a multidimensional regression ($C^{(5)} > 1$), so we define $y_{ic}^{(5)}$ as the share of class c in sample i, and optimize the Cross-Entropy (CE) loss

$$\mathcal{L}^{(5)} = \sum_{c=1}^{C^{(5)}} -y_{ic}^{(5)} \log \tilde{y}_{ic}^{(5)}, \tag{3}$$

with

$$\tilde{y}_{ic}^{(5)} = \text{Softmax}(\hat{y}_{ic}^{(5)}). \tag{4}$$

3.3.4 Parameter Update. All parameters $\theta = \{\theta_{\text{ENC}}, \theta_{\text{REC}}^{(1)}, \ldots, \theta_{\text{REC}}^{(T)}\}$ are trained by backpropagation of the total loss

$$\mathcal{L} = \sum_{t=1}^{T} \lambda_t \mathcal{L}^{(t)}. \tag{5}$$

with λ_t being the loss-weight for reconstruction task t. This leads to the reconstruction models being updated only w.r.t. the task-individual loss, as

$$\nabla_{\theta_{REC}^{(t)}} \mathcal{L}^{(i)} = 0, \text{ if } i \neq t \tag{6}$$

hence

$$\nabla_{\theta_{REC}^{(t)}} \mathcal{L} = \lambda_t \nabla_{\theta_{REC}^{(t)}} \mathcal{L}^{(t)} \tag{7}$$

Further, the encoder model is updated with respect to all task-individual loss terms, as

$$\nabla_{\theta_{ENC}} \mathcal{L} = \nabla_{\theta_{ENC}} \sum_{t=1}^{T} \lambda_t \mathcal{L}^{(t)} = \sum_{t=1}^{T} \lambda_t \nabla_{\theta_{ENC}} \mathcal{L}^{(t)} \tag{8}$$

3.4 Hyperparameter Settings

For our study, we use a bounding box near a medium-sized city[3], as this location covers a variety of road features such as motorways, rural roads, or junctions in urban areas. We train our model for a maximum of 100 epochs with early stopping based on the validation loss. The encoder consists of a stack of 6 Transformer encoders, each with 32 head Attention layers. The loss weights λ_t are all set to 1. Further we train, 4 different versions of *Route2Vec* with 4 different embedding space sizes *i.e.*, Small ($D = 128$), Medium ($D = 256$), Large ($D = 512$), and Extra-large ($D = 1024$).

4 Results

4.1 Semantic Exploration

4.1.1 Cluster Analysis. In order to verify that routes with similar road contexts are encoded to similar embeddings, we visualize the characteristics of routes with similar embeddings. To these means, we randomly select an embedding from the embedding space and retrieve its 9 nearest neighbors. Figure 5 (a) - (c) visualizes certain characteristics of the so-found routes *i.e.*, the thicknesses of the bars indicate the driveable speed, the lengths of the segments correspond to the travel-time, and the colors represent the road type. We can observe that neighboring encoded routes have similar characteristics *e.g.*, in Figure 5c all of them predominantly consist of segments with road type *secondary* and have similar characteristics in terms of speed. By comparing Figure 5 (a) and (b), we can additionally observe that the use of positional embedding vectors leads to the road-context embeddings not only encoding information about the overall presence of characteristics in the sequence but also about their position in the sequence. In Figure 5d, we compare 10 randomly selected encoded routes. The figure shows the variety of characteristics the sequences cover and underlines the usefulness of the projection function learned by *Route2Vec*.

[3] Stuttgart, Germany

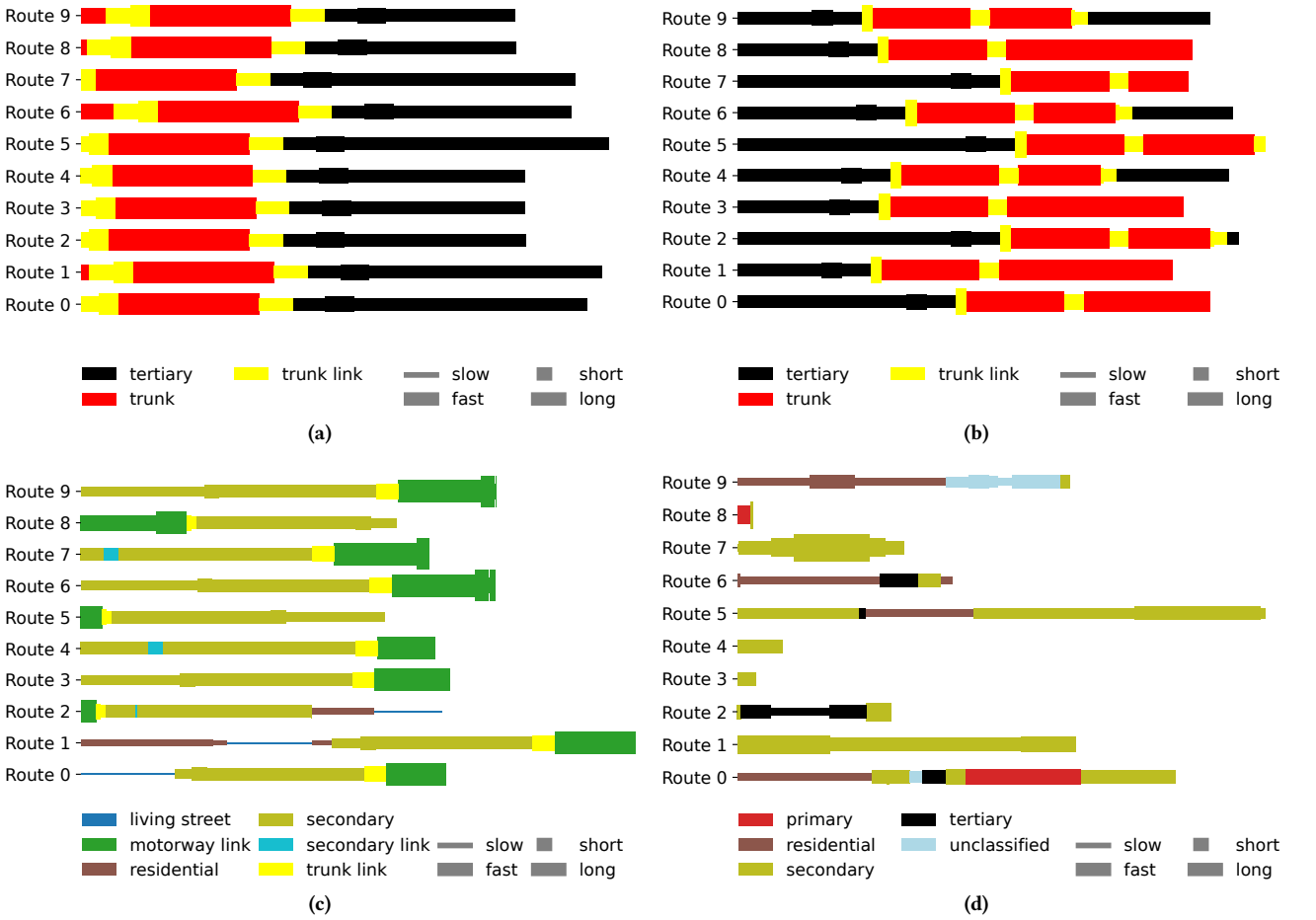


Figure 5: The visualization shows characteristics of roads that *Route2Vec* has computed similar embeddings for. (a) - (c) each depicts a distinct route (Route 0) along with the nine routes whose embeddings are closest in the embedding space. (d) shows a randomly selected set of routes in the embedding space. One can see that embeddings neighboring in the embedding space represent routes with similar road contexts.

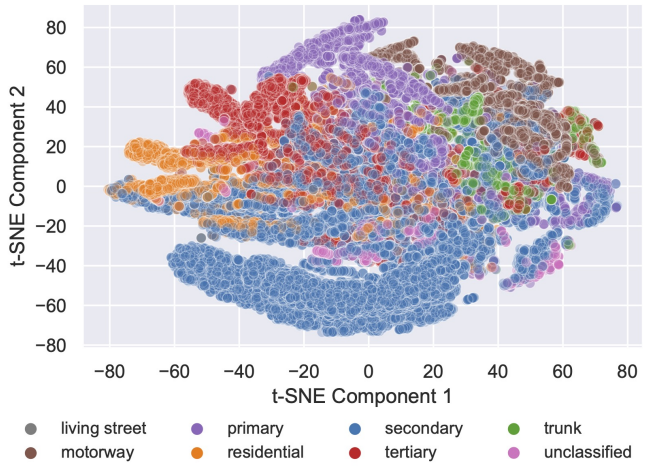
4.1.2 Embedding Space Visualization. In Figure 6, we map the 512-dimensional embedding space of *Route2Vec*-Large into a 2-dimensional space by using t-distributed stochastic neighbor embedding (t-SNE) [31]. In Figure 6a, the samples are color-coded by their predominant road type, *i.e.* by the road type with the highest share throughout the sequence. We observe a clustering for different predominant road types. Further, in Figure 6b, the same embedding space is color-coded by the share of the road type *secondary*. We can identify a separate cluster of samples with a high secondary road share. Samples with a slightly lower share of this road type are encoded to slightly different embeddings, visible next to the clusters with a high share. Both figures indicate that routes of similar road contexts are mapped to similar embeddings.

4.2 Linear Probing

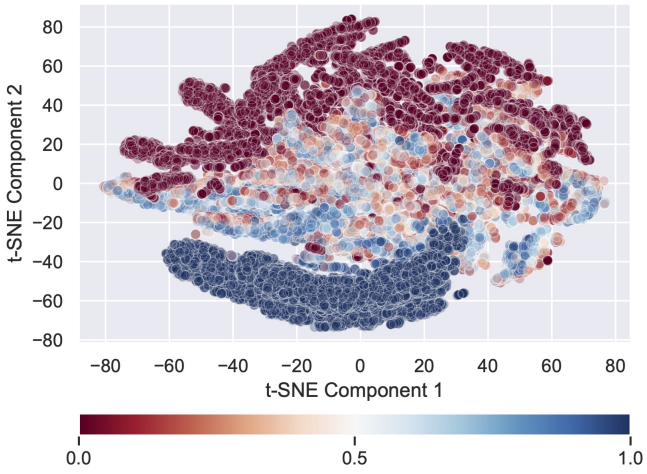
For a quantitative evaluation of the learned embeddings, we employ a linear probing schema, *i.e.*, we freeze the encoder network's parameters and train multiple linear models to reconstruct certain contextual characteristics from the learned embeddings. We define the following labels: (1) Summed Travel Time, (2) Summed Route Length, (3) Mean Curvature, and (4) Mean Absolute Grade. Each reconstruction model is randomly initialized and trained for 100 epochs. We report the adjusted R^2-Score ($\overline{R}^2$) and the mean squared error (*MSE*) on the testing data; the results are shown in Table 2. The labels for tasks (1) and (2) are sums of normalized features, whereas the labels for tasks (3) and (4) are mean values of normalized features. Therefore, the *MSE* for tasks (1) and (2) are, as expected, higher than for tasks (3) and (4). We will first focus on the results of the *Route2Vec* Large configuration with 512-dimensional embeddings. The $\overline{R}^2$-Scores of 0.78, 0.75, and 0.96 together with the *MSE*s of 1.06, 1.91, and < 0.01 respectively, evaluated for the tasks (1), (2), and (4) indicate that variance related to (1) the summed travel time, (2) the summed route length, and (4) the mean absolute grade is encoded in the learned embeddings. Even

Table 2: Quantitative evaluation results of the reconstruction tasks for different embedding dimensions D.

Representation Evaluation Task	Small $D = 128$ 2.4M params $\overline{R}^2$ (↗)	Small MSE (↘)	Medium $D = 256$ 5.3M params $\overline{R}^2$ (↗)	Medium MSE (↘)	Large $D = 512$ 12.6M params $\overline{R}^2$ (↗)	Large MSE (↘)	Extra-large $D = 1024$ 33.6M params $\overline{R}^2$ (↗)	Extra-large MSE (↘)
(1) Summed Travel Time Regression	0.58	2.11	0.71	1.42	0.78	1.06	0.81	0.87
(2) Summed Route Length Regression	0.53	3.70	0.71	2.23	0.75	1.91	0.78	1.61
(3) Mean Curvature Regression	< 0.01	< 0.01	< 0.01	< 0.01	< 0.01	< 0.01	< 0.01	< 0.01
(4) Mean Absolute Grade Regression	0.71	0.01	0.90	< 0.01	0.96	< 0.01	0.97	< 0.01



(a) predominant road type



(b) road type share of class secondary

Figure 6: A t-SNE visualization of the embeddings, color-coded by different characteristics, reveals that routes with similar road contexts are encoded to similar embeddings.

though the $\overline{R}^2$-Score for the task (3) is < 0.01, the corresponding *MSE* of < 0.01 shows that this may be due to the low variance of the label. Thus, both values combined indicate that the evaluation model can predict the label from the embeddings.

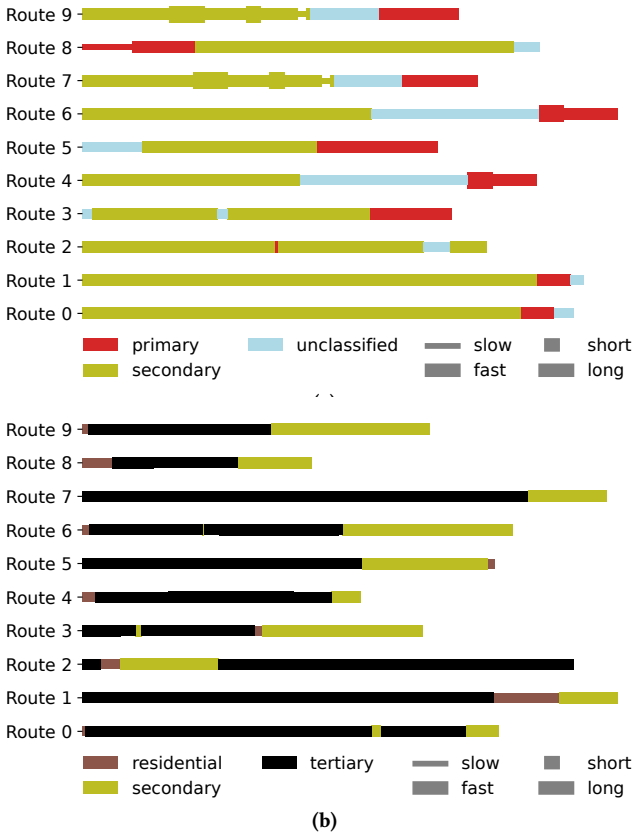
4.3 Influence of the Dimensionality in the Embedding Space

We evaluate the influence of the dimensionality of the learned embeddings by training different versions of *Route2Vec* with different embedding sizes (see Table 2). Overall, we report an expected decrease in performance if using lower-dimensional embeddings, *e.g.* for the summed travel time regression (1), we observe an increase in *MSE* from 1.06 (Large) to 2.11 (Small). Further, we observe an increase in the evaluation models' performances if the embedding size is increased to $D = 1024$, *e.g.*, *MSE* for (1) the summed travel time decreases, and $\overline{R}^2$ does increase from 0.78 to 0.81.

4.4 Generalization to Unseen Regions

We evaluate how meaningful the embeddings of *Route2Vec* are for routes from regions with other regional peculiarities than the training data. Thus, we evaluate *Route2Vec*'s capabilities to generalize to unseen data. Therefore, we sample a second dataset according to the methodology proposed in Section 3.2, this time with a bounding box from a different country[4]. Additionally, we sample only 10% as many routes as for the training dataset and increase the area covered by the bounding box by a factor of 4 to make the presence of similar routes less likely ($N_S = 1000$, and area $\approx 5730\ km^2$). We analyze the embedding space by visualizing clusters of similar embeddings, shown in Figure 7. As expected, we observe that routes from the unseen region that form a cluster in the embedding space are less similar than the ones previously found (Figure 5). Still, we can report that these routes share similar characteristics *e.g.* regarding road types and speed profile. The decrease in similarity is likely due to the absence of more similar routes in the dataset; hence, the embeddings forming a cluster eventually still represent routes of highest similarity in the dataset. Thus, the experiment provides solid foundations that the trained version of *Route2Vec* can be used as a road context-extractor for data of arbitrary regions without laborious fine-tuning.

[4]Florence, Italy



(b)

Figure 7: The visualization shows features of routes for which *Route2Vec* has computed similar embeddings. The corresponding routes were sampled from a region different from the training data. Thus, they have different regional peculiarities.

5 Discussion

5.1 Limitations

Interpretability of Embeddings. While *Route2Vec* effectively encodes contextual similarities between routes, the resulting embeddings are inherently abstract and not directly interpretable by humans. This limits the transparency of the system and poses challenges for applications where explainability is critical, such as in safety-critical mobility systems. For example, although two routes may have similar embeddings, it is not immediately clear which specific contextual features contribute to their proximity in the embedding space. While our evaluation tasks provide indirect evidence of which aspects are captured, a more interpretable or disentangled embedding structure could support its trustworthiness. Future work may explore methods such as post-hoc explanation tools to provide greater insight into the semantics of the learned embedding space.

Sensitivity to Regional Peculiarities. We conducted extensive qualitative and quantitative analyses to examine the embedding space learned by *Route2Vec*. A set of four representation evaluation tasks revealed that road contextual information is quantitatively encoded in the embeddings. The qualitative analyses of the learned embedding space confirmed that routes with similar context sets are encoded in close proximity to the embedding space. Still, we find the learned embedding space to depend on the regional peculiarities in road characteristics of the bounding box from which the routes were sampled. However, the used feature set is independent of such regions. Thus, while features like curvature might become more important if, for example, routes from a mountainous region are used solely for training *Route2Vec*, *Route2Vec* will still be able to encode routes from non-mountainous regions. Further, we demonstrated in our analyses that the key property of *Route2Vec*, i.e., encoding similar routes to similar embeddings, still holds if routes of such unseen and different environments are used.

Dynamic Context Features. This work focused on static road contexts, such as road type distributions, curvature, and intersections, which are inherently reusable across different times and scenarios. This focus enables the embeddings produced by *Route2Vec* to be encoded once and reused later on. However, dynamic context variables such as weather, traffic conditions, or time of day might also be relevant for real-world applications, as they influence both the driving experience and the preferences of drivers and passengers. In future work, we aim to incorporate such dynamic context into *Route2Vec* to create even richer embeddings. One potential approach is to include these variables directly as features during the training process alongside static route context. Alternatively, dynamic context variables could be encoded separately and concatenated with static embeddings, creating a modular framework that integrates both static and dynamic factors [24]. This extension would enable *Route2Vec* to support dynamic context-aware systems that respond to the full range of factors influencing driving conditions.

5.2 Building In-the-wild Context-Aware Systems with *Route2Vec*

Route2Vec encodes heterogeneous driving context into fixed-size semantic embeddings, offering a compact and expressive representation of a route. This makes it a powerful foundation for building real-world context-aware systems.

Building such systems based on *Route2Vec* involves three main steps

(1) **Capture the route context**, either via onboard sensors (e.g., cameras) or by querying contextual map APIs as done in this work.
(2) **Encode the context into an embedding** using a pretrained *Route2Vec* model. If using the same context variables as in this work, the data can be used directly. If new variables are introduced, either a supplementary model is needed, or a new *Route2Vec* model must be trained as outlined in this work.
(3) **Feed the embedding into a downstream system**, that was trained to perform specific tasks, such as predicting drivers' interruptibility [36] or anticipating infotainment interactions [35]. The same embedding can support multiple downstream tasks in parallel.

This modular architecture, with *Route2Vec* as a shared feature extractor, offers three key benefits over end-to-end models. First, it would significantly reduce the amount of raw contextual data that contextual systems need to process while retaining meaningful semantic information. Second, the embeddings could be shared across various context-aware downstream systems, preventing computationally costly task-specific and redundant feature extraction in each of the systems separately. And third, the embeddings could be directly used to identify route segments with contexts similar to those previously observed. Thus, *Route2Vec* can serve as the backbone for real-time, context-aware interactions that enhance the adaptability and personalization of mobile systems, particularly in dynamic driving environments.

5.3 Applications in Human-Computer Interaction

Route2Vec's ability to encode variable-length sequences of mixed-valued feature vectors into a single embedding vector enables easy evaluation of quantitative similarities between road-contextualized routes. We are convinced that *Route2Vec* will be useful for various application domains in Human-Computer Interaction due to this property.

For example, *Route2Vec* can be used to build **smart in-vehicle Human-Machine Interfaces (HMIs)** that adapt shown content dynamically to the route context drivers are currently exposed to. For example, if a driver is entering a high-traffic urban environment, the HMI could proactively activate a calming playlist or propose alternate routes. When combining route context embeddings with user profiles and historical interaction data in a context-aware recommender system [16], this could significantly reduce interaction effort, surface timely content, and ultimately contribute to a more seamless and supportive in-vehicle experience.

Further, *Route2Vec*'s capabilities to quantitatively compare routes by their context could also be leveraged for **retrieval tasks**. Given the embedding of one route, one can easily find routes with similar sets of route context by searching for the closest neighbors in the embedding space of *Route2Vec*. This would allow, for example, novel fitness applications to suggest running, cycling, or walking routes that match users' historical preferences or mimic the contextual features of iconic tours, such as the elevation profiles or road types of the Alpe d'Huez. Similarly, leisure-oriented navigation apps could suggest scenic "Sunday drives" that reflect a user's environmental preferences, like winding rural roads. Thus, *Route2Vec* would allow route planning to go beyond time or distance by incorporating context-oriented criteria. Additionally, developers for autonomous driving functions could use the embedding space to search for and curate collections of routes with similar and potentially challenging characteristics, such as sequences of sharp turns [2, 14, 33]. This would allow for more targeted testing and validation of autonomous systems under comparable contextual conditions without needing to manually label or categorize massive route datasets.

Finally, in **next Point-of-Interest (POI) recommender systems** [13], the fixed-size embeddings output by *Route2Vec* might serve as context-aware item embeddings. Hence, rather than recommending POIs solely based on previous visits or content similarity, systems could incorporate the contextual features of routes that users have taken in the past. For instance, a driver who often travels scenic rural roads might prefer POIs like lookout points, picnic areas, or serene coffee shops, whereas a commuter navigating urban highways might be more interested in quick-service restaurants or fuel stations. Embedding the route context allows the system to align POI suggestions with the environments users tend to enjoy, creating more meaningful and situationally relevant recommendations.

Thus, *Route2Vec* has the potential to enable or facilitate a variety of different applications. While the ideas hold promise, further research is needed to assess whether *Route2Vec*'s embeddings capture sufficient nuances of route context for these applications.

6 Conclusion

With *Route2Vec*, we introduce a novel semantic road-context embedding system. We demonstrated *Route2Vec*'s ability to learn road-context embeddings from routes of varying lengths in a self-supervised learning schema. Our analyses show that similar sets of road context are encoded to similar representations, allowing for easy comparison and retrieval of contextual routes. We are confident that our work will advance the field of context-aware systems by providing a plug-and-play module that enables the use of contextual road features. This capability can enable context-informed interactions with intelligent systems in moving platforms, such as adaptive route planning, personalized infotainment, and proactive driver assistance. By enabling efficient comparisons between sets of route context, we are confident that *Route2Vec* can become a foundational building block for developing advanced context-aware applications.

Acknowledgments

This work is supported by the German Research Foundation (DFG), CRC 1404: "FONDA: Foundations of Workflows for Large-Scale Scientific Data Analysis" (Project-ID 414984028).

References

[1] Alexei Baevski, Wei-Ning Hsu, Qiantong Xu, Arun Babu, Jiatao Gu, and Michael Auli. 2022. Data2vec: A general framework for self-supervised learning in speech, vision and language. *arXiv preprint arXiv:2202.03555* (2022).

[2] Mohammadhossein Bahari, Saeed Saadatnejad, Ahmad Rahimi, Mohammad Shaverdikondori, Amir Hossein Shahidzadeh, Seyed-Mohsen Moosavi-Dezfooli, and Alexandre Alahi. 2022. Vehicle trajectory prediction works, but not everywhere. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 17123–17133.

[3] David Bethge, Daniel Bulanda, Adam Kozlowski, Thomas Kosch, Albrecht Schmidt, and Tobias Grosse-Puppendahl. 2024. HappyRouting: Learning Emotion-Aware Route Trajectories for Scalable In-The-Wild Navigation. *arXiv preprint arXiv:2401.15695* (2024).

[4] David Bethge, Luis Falconeri Coelho, Thomas Kosch, Satiyabooshan Murugaboopathy, Ulrich von Zadow, Albrecht Schmidt, and Tobias Grosse-Puppendahl. 2023. Technical design space analysis for unobtrusive driver emotion assessment using multi-domain context. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 6, 4 (2023), 1–30.

[5] David Bethge, Thomas Kosch, Tobias Grosse-Puppendahl, Lewis L Chuang, Mohamed Kari, Alexander Jagaciak, and Albrecht Schmidt. 2021. VEmotion: Using Driving Context for Indirect Emotion Prediction in Real-Time. In *The 34th Annual ACM Symposium on User Interface Software and Technology*. 638–651.

[6] David Bethge, Constantin Patsch, Philipp Hallgarten, and Thomas Kosch. 2023. Interpretable Time-dependent convolutional emotion recognition with contextual data streams. In *Extended Abstracts of the 2023 CHI Conference on Human Factors in Computing Systems*. 1–9.

[7] Geoff Boeing. 2017. OSMnx: New methods for acquiring, constructing, analyzing, and visualizing complex street networks. *Computers, Environment and Urban Systems* 65 (2017), 126–139.

[8] Corine Land Cover. 2018. European Union, Copernicus Land Monitoring Service 2018. *European Environment Agency (EEA)* (2018).

[9] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805* (2018).

[10] M Fathi and MR Masnavi. 2014. Assessing environmental aesthetics of roadside vegetation and scenic beauty of highway landscape: preferences and perception of motorists. *International Journal of Environmental Research* 8, 4 (2014), 941–952.

[11] Anna-Katharina Frison, Philipp Wintersberger, Tianjia Liu, and Andreas Riener. 2019. Why do you like to drive automated? a context-dependent analysis of highly automated driving to elaborate requirements for intelligent user interfaces. In *Proceedings of the 24th international conference on intelligent user interfaces*. 528–537.

[12] Philipp Hallgarten, David Bethge, Ozan Özdcnizci, Tobias Grosse-Puppendahl, and Enkelejda Kasneci. 2023. TS-MoCo: Time-Series Momentum Contrast for Self-Supervised Physiological Representation Learning. In *2023 31st European Signal Processing Conference (EUSIPCO)*. IEEE, 1030–1034.

[13] Daniel Herzog, Sherjeel Sikander, and Wolfgang Wörndl. 2019. Integrating route attractiveness attributes into tourist trip recommendations. In *Companion Proceedings of The 2019 World Wide Web Conference*. 96–101.

[14] Bo Jiang, Shaoyu Chen, Qing Xu, Bencheng Liao, Jiajie Chen, Helong Zhou, Qian Zhang, Wenyu Liu, Chang Huang, and Xinggang Wang. 2023. Vad: Vectorized scene representation for efficient autonomous driving. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 8340–8350.

[15] SeungJun Kim, Jaemin Chun, and Anind K Dey. 2015. Sensors know when to interrupt you in the car: Detecting driver interruptibility through monitoring of peripheral interactions. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*. 487–496.

[16] Saurabh Kulkarni and Sunil F Rodd. 2020. Context Aware Recommendation Systems: A review of the state of the art techniques. *Computer Science Review* 37 (2020), 100255.

[17] Jean-Christophe Léger. 1999. Menger curvature and rectifiability. *Annals of mathematics* 149, 3 (1999), 831–869.

[18] Shu Liu, Kevin Koch, Zimu Zhou, Simon Föll, Xiaoxi He, Tina Menke, Elgar Fleisch, and Felix Wortmann. 2021. The Empathetic Car: Exploring Emotion Inference via Driver Behaviour and Traffic Context. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 5, 3, Article 117 (Sept. 2021), 34 pages. doi:10.1145/3478078

[19] Gengchen Mai, Krzysztof Janowicz, Bo Yan, Rui Zhu, Ling Cai, and Ni Lao. 2020. Multi-scale representation learning for spatial feature distributions using grid cells. *arXiv preprint arXiv:2003.00824* (2020).

[20] Arnav Vaibhav Malawade, Shih-Yuan Yu, Brandon Hsu, Harsimrat Kaeley, Anurag Karra, and Mohammad Abdullah Al Faruque. 2022. Roadscene2vec: A tool for extracting and embedding road scene-graphs. *Knowledge-Based Systems* 242 (2022), 108245.

[21] Bruce Mehler, Bryan Reimer, Lisa A D'Ambrosio, Alexander Piña, and Joseph F Coughlin. 2010. An Evaluation of Time of Day Influences on Simulated Driving Performance and Physiological Arousal. In *Proceedings of the 89th Annual Meeting of the Transportation Research Board on Traffic and Transport Planning*. 1–15.

[22] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781* (2013).

[23] OpenStreetMap contributors. 2017. Street Graph dump retrieved from https://planet.osm.org . https://www.openstreetmap.org.

[24] Marco Polignano, Cataldo Musto, Marco de Gemmis, Pasquale Lops, and Giovanni Semeraro. 2021. Together is Better: Hybrid Recommendations Combining Graph Embeddings and Contextualized Word Representations. In *Fifteenth ACM Conference on Recommender Systems*. 187–198.

[25] Nina Runge, Pavel Samsonov, Donald Degraen, and Johannes Schöning. 2016. No more autobahn! Scenic route generation using Googles street view. In *Proceedings of the 21st International Conference on Intelligent User Interfaces*. 147–151.

[26] Albrecht Schmidt, Michael Beigl, and Hans-W Gellersen. 1999. There is more to context than location. *Computers & Graphics* 23, 6 (1999), 893–901.

[27] Stefan Schneegaß, Bastian Pfleging, Dagmar Kern, and Albrecht Schmidt. 2011. Support for modeling interaction with automotive user interfaces. In *Proceedings of the 3rd International Conference on Automotive User Interfaces and Interactive Vehicular Applications* (Salzburg, Austria) *(AutomotiveUI '11)*. Association for Computing Machinery, New York, NY, USA, 71–78. doi:10.1145/2381416.2381428

[28] Rob Semmens, Nikolas Martelaro, Pushyami Kaveti, Simon Stent, and Wendy Ju. 2019. Is Now A Good Time? An Empirical Study of Vehicle-Driver Communication Timing. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*. 1–12.

[29] Arash Tavakoli, Vahid Balali, and Arsalan Heydarian. 2020. A Multimodal Approach for Monitoring Driving Behavior and Emotions. Mineta Transportation Institute.

[30] Arash Tavakoli, Mehdi Boukhechba, and Arsalan Heydarian. 2020. Personalized Driver State Profiles: A Naturalistic Data-Driven Study. In *Proceedings of the AHFE 2020 Virtual Conference on Human Aspects of Transportation*. Springer, 32–39.

[31] Laurens van der Maaten and Geoffrey Hinton. 2008. Visualizing Data using t-SNE. *Journal of Machine Learning Research* 9, 86 (2008), 2579–2605.

[32] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems* 30 (2017).

[33] Junyao Wang, Arnav Vaibhav Malawade, Junhong Zhou, Shih-Yuan Yu, and Mohammad Abdullah Al Faruque. 2024. Rs2g: Data-driven scene-graph extraction and embedding for robust autonomous perception and scenario understanding. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*. 7493–7502.

[34] Marco Wiedner, Sreerag V Naveenachandran, Philipp Hallgarten, Satiyabooshan Murugaboopathy, and Emilio Frazzoli. 2024. CARSI II: A Context-Driven Intelligent User Interface. In *Adjunct Proceedings of the 16th International Conference on Automotive User Interfaces and Interactive Vehicular Applications*. 128–135.

[35] Jannik Wolf, Marco Wiedner, Mohamed Kari, and David Bethge. 2021. HMInference: Inferring multimodal HMI interactions in automotive screens. In *13th international conference on automotive user interfaces and interactive vehicular applications*. 230–236.

[36] Tong Wu, Nikolas Martelaro, Simon Stent, Jorge Ortiz, and Wendy Ju. 2021. Learning when agents can talk to drivers using the INAGT dataset and multisensor fusion. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 5, 3 (2021), 1–28.