

Article

Not peer-reviewed version

Research on Driving Scenario Knowledge Graphs

[Ce Zhang](#)^{*}, Liang Hong, Dan Wang, [Xinchao Liu](#), Jinzhe Yang, [Yier Lin](#)^{*}

Posted Date: 3 April 2024

doi: 10.20944/preprints202404.0271.v1

Keywords: intelligent traffic; knowledge graph; hybrid methods; driving scenarios; ontology



Preprints.org is a free multidiscipline platform providing preprint service that is dedicated to making early versions of research outputs permanently available and citable. Preprints posted at Preprints.org appear in Web of Science, Crossref, Google Scholar, Scilit, Europe PMC.

Copyright: This is an open access article distributed under the Creative Commons Attribution License which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Article

Research on Driving Scenario Knowledge Graphs

Ce Zhang, Liang Hong, Dan Wang, Xinchao Liu, Jinzhe Yang and Yier Lin *

College of Mechanical Engineering, Tianjin University of Science and Technology, Tianjin 300222, China; zhangce@mail.tust.edu.cn (C.Z.)

* Correspondence: yier.lin@tust.edu.cn; Tel.: +86-185-2213-3115

Abstract: Despite the partial disclosure of driving scenario knowledge graphs, they still fail to meet the comprehensive needs of intelligent connected vehicles for driving knowledge. Current issues include the high complexity of pattern layer construction, insufficient accuracy of information extraction and fusion, and limited performance of knowledge reasoning models. To address these challenges, a hybrid knowledge graph method was adopted in the construction of the Driving Scenario Knowledge Graph (DSKG). Firstly, core concepts in the field were systematically sorted and classified, laying the foundation for the construction of a multi-level classified knowledge graph top-level ontology. Subsequently, by deeply exploring and analyzing the Traffic Genome data, 34 entities and 51 relations were extracted and integrated with the ontology layer, achieving the expansion and updating of the knowledge graph. Then, in terms of knowledge reasoning models, an analysis of the training results of the TransE, Complex, Distmult, and Rotate models in the entity linking prediction task of DSKG revealed that the Distmult model performed the best in metrics such as hit rate, making it more suitable for inference in DSKG. Finally, a standardized and widely applicable Driving Scenario Knowledge Graph was proposed. The DSKG and related materials have been publicly released for use by industry and academia.

Keywords: intelligent traffic; knowledge graph; hybrid methods; driving scenarios; ontology

1. Introduction

In recent years, significant progress has been made in the field of autonomous driving, largely attributed to the development and application of artificial intelligence technologies, particularly algorithms such as deep learning and reinforcement learning. These algorithms play crucial roles in perception, decision-making, and control, enabling vehicles to accurately perceive and adapt to their surrounding environments.

Despite remarkable achievements, the comprehensive application of advanced autonomous driving technologies still faces numerous challenges. This is mainly due to the limitations of artificial intelligence algorithms in interpretability, understanding complex scenarios, and handling unknown situations [1]. When faced with real-world situations that differ from the training data, current artificial intelligence algorithms may encounter significant setbacks, potentially leading to accidents involving autonomous vehicles. Additionally, artificial intelligence algorithms rely on vast training datasets, requiring them to process large volumes of example data to achieve high accuracy, which sharply contrasts with the efficiency of humans in integrating new information through single instances of learning. In this context, understanding context becomes particularly important for agents to handle unknown situations, provide feedback to users, and understand their own functionality.

Scene understanding requires systems to possess a profound semantic understanding of entities and their relationships in complex physical and social environments. To meet this requirement, a feasible approach is to represent entities and their relationships in scenes as knowledge graphs (KGs) [2]. By constructing scene knowledge graphs, systems can better predict entity behaviors, thereby enhancing scene understanding capabilities. Therefore, knowledge graph construction based on

knowledge injection learning holds tremendous potential in addressing the complex technical challenges of scene understanding in autonomous driving.

However, the application of knowledge graphs also faces a series of challenges and issues, such as the cost of knowledge acquisition, the uncertainty of knowledge representation, and the timeliness of knowledge updates. Among these, the cost of knowledge acquisition is a significant consideration factor because building and maintaining a complete and accurate knowledge graph require substantial human and time investments. Additionally, the uncertainty of knowledge representation poses another challenge because real-world knowledge is often fuzzy, uncertain, and may evolve over time and in different environments. There is also a high demand for timely knowledge updates because autonomous driving systems need to promptly acquire the latest information to adapt to continuously changing road conditions and traffic rules.

The main contributions of this paper include: Firstly, proposing a more efficient method for knowledge acquisition and updating, including automated extraction and updating of knowledge from various sources to reduce manpower and time costs; Secondly, constructing a standardized and widely applicable knowledge graph for driving scenarios; Finally, validating the inference effects of different knowledge embedding models in DSKG to discover new knowledge and confirm the most effective embedding models for scene understanding.

The subsequent sections of this paper will unfold in the following order: Section 2 will review the construction methods of knowledge graphs; Section 3 will detail the construction process of DSKG and provide in-depth analysis and discussion of all evaluation results; Finally, Section 4 will summarize the main conclusions of this paper and outline future research directions.

2. Knowledge Graph

The concept of Knowledge Graph [3] was formally introduced by Google in 2012, aiming to depict various concepts, entities, and their relationships in the real world using graph theory. In this structure, entities are interconnected through relationships, forming a networked knowledge system. Compared to traditional text and tabular representations, Knowledge Graph, with its intuitive and efficient characteristics, can clearly display entity attributes and their connections, thus enabling tasks such as intelligent search, question reasoning, and recommendations.

A Knowledge Graph mainly consists of two layers: the schema layer and the data layer. The ontology, as the schema layer of the Knowledge Graph, constrains the data layer through rules and axioms defined by the ontology. The data layer, situated below the ontology layer, stores information in the form of triple tables, attribute tables, etc., in a graph database. For the schematic diagram of a domain knowledge graph structure, please refer to Figure 1.

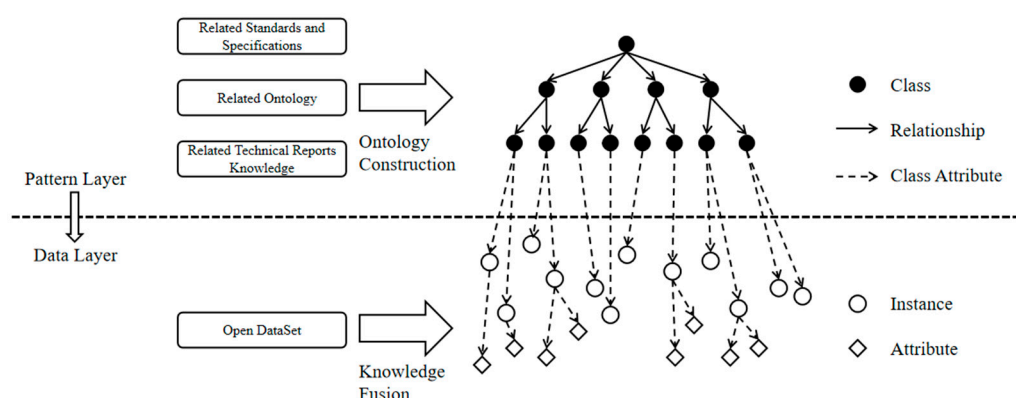


Figure 1. Conceptual representation of knowledge graph structure.

The construction of a Knowledge Graph can be mainly divided into two categories: top-down and bottom-up approaches. The top-down approach establishes the schema layer first, guiding the construction of the data layer, directly extracting ontology and schemas from structured data sources,

and incorporating them into the knowledge base. This method performs well in displaying hierarchical relationships between concepts but requires a high degree of human involvement and has limited schema layer updates, making it more suitable for constructing knowledge graphs with smaller scales and clear hierarchical structures. If a domain knowledge graph has a small scale and a clear knowledge system, the top-down approach can be considered. The bottom-up approach, on the other hand, first constructs the data layer, integrates the acquired knowledge, extracts and filters ontology patterns from the data, and then manually selects new patterns with high confidence to add to the knowledge base. This method updates rapidly and is suitable for large-scale knowledge graph construction but may contain more noise. Typically, the bottom-up approach is chosen for constructing general-purpose knowledge graphs due to the richness and complexity of general knowledge, which makes defining patterns manually challenging.

To balance the advantages of both approaches, this paper adopts a hybrid knowledge graph construction approach. The basic strategy of hybrid knowledge graph construction is as follows: Firstly, through literature research and reading specifications, the domain concept system is summarized, and the knowledge architecture is extended in layers to obtain a multidimensional hierarchical classification knowledge system, and then construct the top-level ontology of the multidimensional hierarchical knowledge graph. Secondly, based on massive structured data, knowledge extraction is carried out to obtain a small-scale graph structure relevant to new knowledge. Finally, after the fusion and update of entities and data, a hybrid structure intelligent connected vehicle driving scenario domain knowledge graph containing both hierarchical and networked systems is constructed.

3. Methods

Addressing the issues of inconsistent concepts and relationships within the domain-specific knowledge graph for intelligent connected vehicle driving scenarios, this chapter proposes a hybrid method for constructing a driving scenario knowledge graph. The overall process is illustrated in Figure 2. Firstly, the method collects core knowledge from the public domain of driving scenarios. Secondly, it constructs a multi-level classified knowledge graph top-level ontology based on open concepts and ontology knowledge, and extracts entities, attributes, and relationships from massive structured data to form a small-scale graph structure. Next, by integrating the schema layer, the knowledge graph for intelligent connected vehicle driving scenarios is constructed. Subsequently, based on existing data in the knowledge graph, knowledge reasoning is performed, and the quality of the reasoning results is evaluated to expand and enrich the knowledge graph. Finally, a fine-grained driving scenario knowledge description space with global and multi-level associations is realized, providing a knowledge foundation for the research on driving scenario graph generation based on visual knowledge fusion in the next chapter.

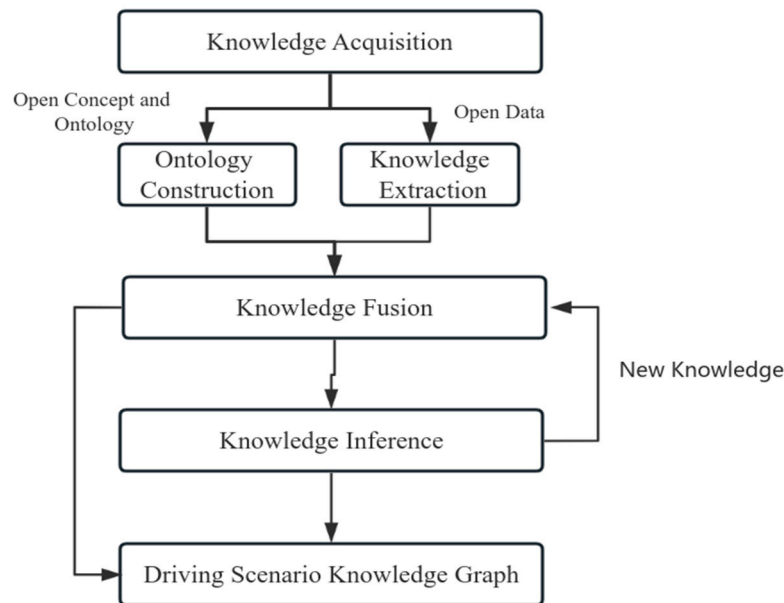


Figure 2. Construction framework of the driving scenario knowledge graph.

3.1. Knowledge Acquisition

The goal of knowledge acquisition is to gather, integrate, and analyze core knowledge in the public domain, thereby constructing a comprehensive and reliable knowledge system. The following will delve into the analysis and discussion of the core knowledge of intelligent connected vehicle driving scenarios collected from the aspects of concepts, ontology, and data.

3.1.1. Domain Concepts

Driving scenario [4] refers to the comprehensive and dynamic description of the interactive process between intelligent connected vehicles and elements in the driving environment such as other vehicles, roads, traffic facilities, and weather conditions within a specific time and space range. It covers the organic combination of driving situations of autonomous vehicles and driving environments, including various entity elements, actions performed by entities, and connections between entities. In Schuldt's related research [5,6], the concept of building scene and environment descriptions using a hierarchical model is proposed. This model consists of four layers: the first layer describes the basic road network, including road markings; the second layer describes specific adjustments to the basic road network, involving the setting of traffic signs, safety facilities, buildings, and streetlights; the third layer describes participants and their controls; and the fourth layer describes environmental conditions. In the PEGASUS project [7], the layered scene concept [6] is applied to highway scenarios. Bagschik et al. [8] and Bock et al. [9] introduced the fifth and sixth layers respectively, forming the widely propagated six-layer model of highway scenarios. In Bagschik et al.'s research [8], the basic road network is divided into two different levels, namely the road layer and the traffic infrastructure layer, to distinguish road descriptions and traffic rules. The definition of the road layer is limited to road layout and its topological structure, while the traffic infrastructure layer includes structural boundaries, traffic signs, and markings. In all subsequent work [10,11], a new third layer is separated from the second layer, focusing on describing temporary changes to the first and second layers. The fourth layer is named "objects," which includes all static, dynamic, and movable objects not belonging to traffic infrastructure. The fifth layer aligns with the definition of environmental conditions in the fourth layer described in Schuldt's paper [6], which is weather conditions. Bock et al. [9] introduced the sixth layer for digital information, used for data and communication information. Considering the emphasis of the PEGASUS project [7] on highway applications, structures along the roadside such as buildings are not elaborated on, and based on the

six-layer model of highway scenarios, Scholtes et al. [12] proposed a six-layer model for urban scenarios. Meanwhile, based on the six-layer external scene construction of the China Industry Innovation Alliance for the Intelligent and Connected Vehicles (CAICV)'s Expected Functional Safety Working Group (referred to as the CAICV-SOTIF Working Group), a seventh layer, namely the vehicle state [13], is added to more accurately describe in-car information. Figure 3 illustrates the iterative process of the scene hierarchical concept evolution.

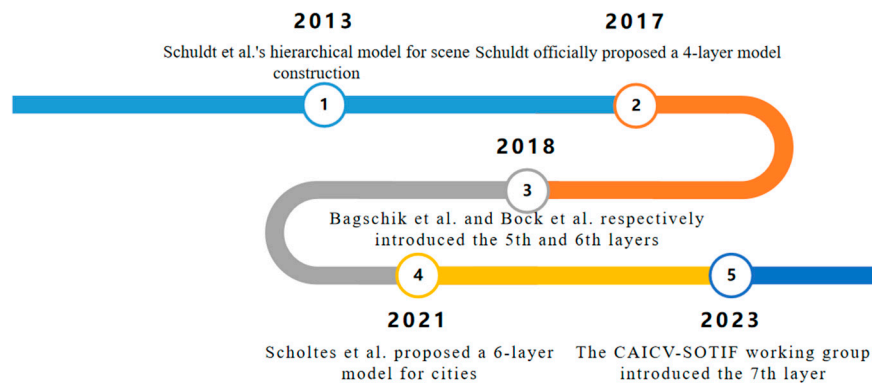


Figure 3. Iterative process of scene hierarchical concept evolution.

3.1.2. Domain Ontology

Ontology is a formalized model for describing domain knowledge, defining semantic associations between entities, properties, and relationships within the domain. In the work of Bagschik et al. [14], an ontology describing simple highway scenes based on a set of predefined keywords was proposed. In subsequent work, Menzel et al. [15] extended this concept to generate OpenSCENARIO and OpenDRIVE scenes. Tahir et al. [16] proposed an ontology focusing on urban intersection scenarios, while also addressing evolving weather conditions. Hermann et al. [17] proposed an ontology for dataset creation, focusing on pedestrian detection, including occlusions of pedestrians. Their ontology, inspired by the Pegasus model [7], consists of 22 sub-ontologies. It can describe various scenarios and convert them into simulations. However, due to the lack of detailed description and unavailability, it is unclear whether a separate ontology is needed for each frame or if the ontology itself can describe temporal scenes. The Association for Standardisation of Automation and Measuring Systems (ASAM) [18] developed the OpenX Ontology, aiming to provide a common definition, attributes, and relationships for central concepts in the ASAM OpenX standard in the field of road traffic. The ontology is mainly divided into three modules: core ontology, domain ontology, and application ontology. The domain ontology defines the central concepts of road traffic, including three layers: environmental conditions, road topology and traffic infrastructure, traffic participants and behaviors. Bogdoll et al. [19] proposed an extensive framework for developing edge hazardous scenarios. Westhofen et al. [20] also proposed an ontology for urban automotive city traffic (A.U.T.O.) to discover more trigger events. Table 1 summarizes the scope and attributes of the previously introduced ontologies, and whether the ontology itself is openly published.

Table 1. Ontology related to driving scenario.

Authors	Year	Scene category	ASAM	Open status
Bagschik et al.	2018	Highway		-
Menzel et al.	2019	Highway		-
Tahir et al.	2022	Urban		-
Hermann et al.	2022	Urban		-
ASAM	2022	Full scene		√

Bogdoll et al.	2022	Full scene	√
Westhofen et al.	2022	Urban	√

3.1.3. Domain Data

In the field of autonomous driving, there are numerous high-quality structured datasets. Here are some commonly used open-source datasets for autonomous driving:

(1) KITTI dataset [21]: Created jointly by the Karlsruhe Institute of Technology in Germany and the Toyota Technical Institute in the United States, it is currently the largest dataset for evaluating computer vision algorithms in autonomous driving scenarios. It includes real image data collected from urban, rural, and highway scenes, as well as data on 3D object detection, tracking, etc.

(2) BDD100K dataset [22]: Released by the Berkeley AI Research Lab (BAIR), it contains 100,000 video frames, with each image having a resolution of 1280x720, and has been annotated. This dataset is mainly used to support research on visual perception in autonomous driving.

(3) NuScenes dataset [23]: A large public dataset for autonomous driving, mainly used for research on tasks such as 3D object detection, tracking, and map construction in autonomous driving.

(4) Waymo dataset [24]: An autonomous driving dataset released by Waymo, a subsidiary of Google, containing a large amount of lidar and camera data, mainly used to support research and development of autonomous driving technology.

(5) A2D2 dataset [25]: A large autonomous driving dataset released by Audi, containing various sensor data such as lidar, cameras, etc., mainly used to support research and development of autonomous driving technology.

(6) PandaSet dataset [26]: Jointly released by domestic lidar manufacturer Hesai Technology and AI data annotation platform company Scale AI, mainly used to support research and development of L5-level autonomous driving technology.

(7) Traffic Genome dataset [27]: A traffic scene graph dataset consisting of 1000 scenes selected from the Urban Scene Dataset [28].

These datasets are rich in autonomous driving scene data and annotation information, providing crucial support for research and development of autonomous driving technology. We analyzed the annotation information of the aforementioned datasets to acquire knowledge and define ontology elements. Table 2 summarizes the number of various elements in each dataset.

Table 2. Datasets related to driving scenario.

Dataset	Year	Number of Concept Classes	Number of Attributes or Relationships
KITTI	2013	5	-
BDD100K	2018	40	-
NuScenes	2019	23	5
A2D2	2019	52	-
Waymo	2020	4	-
PandaSet	2020	37	13
Traffic Genome	2021	34	51

3.2. Construction of Ontology for Driving Scenarios

Constructing the domain ontology for driving scenarios involves considering the standardization of domain terms, the broad applicability of conceptual categories, the hierarchical structure of abstract concepts in the domain, and defining the relevant attributes of each concept and the relationships between concepts [29]. First, existing domain ontologies were examined. Among them, the terminology of the A.U.T.O. ontology, open-sourced by Westhofen et al. [20], was more standardized. Therefore, based on the reuse of the A.U.T.O. ontology, further extensions and adjustments were made for a full-scenario perspective based on the domain knowledge obtained in

this chapter. Additionally, during the subsequent knowledge extraction from structured data, the ontology was updated based on the fusion results of the small-scale knowledge graph, making it not only conform to the domain consensus but also adaptable for describing and representing full-scenario knowledge.

3.2.1. Definition of Classes and Their Hierarchical Structure

Referring to the widely applicable six-layer scene hierarchical concept, this chapter also divides the knowledge system of intelligent connected vehicle driving scenarios into six concept classes: Basic Road, Roadside Facility, Temporary Change, Dynamic Object, Environmental Condition, and Digital Information. Among them, the Basic Road provides the foundational support for all scenario elements; the Roadside Facility is built upon the road structure; the Temporary Change describes transient adjustments within the Basic Road and Roadside Facility ; the Dynamic Object acts as an intermediate layer, connecting static and dynamic scenario elements; the Environmental Condition characterizes the environmental elements of the scenario and their impacts on the aforementioned concept and vehicle functions; and the Digital Information describes all digital data-based information related to vehicles, infrastructure, or both. These concept classes are constructed hierarchically from the bottom layer to the top layer, forming a complete architecture. Each concept class contains specific subclasses, forming a parent-child inheritance relationship among them. The concept classes and their associated attributes together construct the process of abstracting from functional scenes, logical scenes to concrete scenes. The following briefly explains the core concept classes of the domain ontology of driving scenarios and their subclasses:

(1) Basic Road : Describes the road network and all permanent objects required for road traffic guidance. It can be divided into Road Topology, Road Surface Condition, and Traffic Markings.

Road Topology can be mapped to the road network and lanes. For complex road topologies such as intersections and roundabouts, they are recorded using a "road network + lanes" approach to avoid redundant descriptions caused by direct classification of road structures. The road network consists of segments and connections, while lanes contain records of actual roads within each segment.

Road Surface Condition is divided into Road Surface Material and Road Surface Irregularity. The material of the road surface describes its physical characteristics, including features of materials such as asphalt, concrete, gravel, etc. Road surface irregularity describes whether the road surface has manholes, potholes, cracks, faults, depressions, speed bumps, etc., affecting driving conditions.

Traffic markings are traffic indications drawn on the road surface, which can be divided into directional markings, prohibition markings, and warning markings based on their functions. Directional markings indicate markings on facilities such as roadways, directions of travel, road edges, and sidewalks. Prohibition markings indicate special regulations such as adherence to, prohibition, and restriction of road traffic, which drivers and pedestrians must strictly adhere to. Warning markings prompt drivers and pedestrians to understand special situations on the road, increase vigilance, and prepare for contingency measures.

(2) Roadside Facility : Describes all static objects usually placed near the road space rather than on the road. These static objects can be further decomposed into Urban Infrastructure, Traffic Infrastructure, Traffic Control Facilities, and Traffic Information Facilities. Urban infrastructure includes buildings, vegetation, streetlights, fire hydrants, etc.; Traffic infrastructure includes fences, tunnels, bridges, etc.; Traffic control facilities include traffic lights and roadblocks; Traffic information facilities include traffic signs and information display screens.

(3) Temporary Change : Describes the non-persistent temporary changes of entities within the Basic Road and Roadside Facility layers in specific scenarios. This layer does not introduce any new entity classes defined in the preceding layers but consists of temporary modifications of elements from layers 1 and 2. This layer categorizes temporary road events into Road Condition Changes and Road Surface Changes. For each type of temporary event, information such as the lane position, starting point, and end point of the event is specified. Based on the description of the first layer, changes in road conditions can be divided into changes in road curvature, slope, coverage, lane width, number, centerline, etc. In addition to these temporary events closely related to the first layer

of road structure, it also includes road surface changes caused by changes in weather conditions, such as dryness, moisture, icing, or reflective road surfaces.

(4) Dynamic Object : Describes dynamic objects in the scenario that affect the occurrence of events. Dynamic objects can be classified from the perspective of vehicles, people, animals, and other objects. The movement of these objects evolves over time and can be described through trajectories.

(5) Environmental Condition : Describes the natural environment in which the traffic scene is located. Weather conditions include weather conditions, temperature, humidity, wind speed, wind direction, visibility, etc. Lighting conditions include the position of light sources, the type of light sources, the intensity of light, the direction of light, and whether there is reflected light, etc.

(6) Digital Information : Describes all information based on digital data between vehicles, infrastructure, or both, including digital signals from information devices such as roadside units and edge computing units. The description of roadside units includes the change status of traffic signs and traffic lights. It is noteworthy that this layer describes the change status of traffic signs and traffic lights, but is limited to the description of changeable information, while the respective objects themselves have already been placed in layer 1. The description of edge computing units includes the change information of perception and control.

3.2.2. Definition of Class Attributes

After the completion of class definitions, the attributes of each class need to be defined. Since these classes have inheritance, subclasses can inherit attributes from parent classes. Therefore, placing these attributes in the most widely applicable application class, close to the top level, facilitates the efficiency of inheritance. Table 3 shows examples of attributes of some classes involved in this chapter.

Table 3. Example of concept class properties in the driving scenario domain.

Classes	Attributes
Lane	Length, width, direction, type, speed limit, etc.
Traffic markings	Type, color, width, length, shape, maintenance status, etc.
Vehicle	Type, brand, color, driving status, etc.
Person	Age group, gender, activity status, etc.
Object	Type, size, color, material, shape, mobility, etc.

3.2.3. Definition of Class Relationships

Based on the framework of the driving scenario domain ontology, the parent-child relationships of the hierarchical concepts of driving scenarios can be clearly defined. These relationships have transitivity, where subclasses can inherit parent class relationships and may also possess new specific relationships. Therefore, parent class relationships can be defined as common attributes, while subclass relationships are defined as specific relationships. The construction of conceptual relationships between classes in the driving scenario domain adopts a top-down approach. The ontology relationships describe the spatial, temporal, and semantic relationships between entities in traffic scenarios. Specifically, as shown in Table 4.

Table 4. Example of relationships in the driving scenario domain.

Relationship Category	Meaning	Examples
Spatial Relationship	Describes the topological, directional, and metric relationships between concept classes.	The position dependency between lanes, road irregularities, and roadside facilities. The positional relationship between the driving direction of the vehicle and dynamic objects such as vehicles and pedestrians.

		The relative distance between the vehicle and roadside facilities, dynamic objects, and other elements.
		Whether the vehicle passes through the intersection during the green light phase of the traffic signal.
		Changes in environmental conditions over time during vehicle travel.
		The duration of vehicle parking in temporary parking areas.
		The restricted access rules for the tidal lane at different times.
Temporal Relationship	Describes the geometric topology information of time points or timelines between concept classes.	
	Describes the traffic connections between concept classes to	
Semantic Relationship	express accessibility or restrictions, subject to constraints of time and space.	The relationship between people and vehicles in terms of driving or being driven.

In this section, we utilize the Protégé ontology management software developed by Stanford University for ontology editing and management. During the construction of the domain ontology, we manually input the relevant ontology of driving scenarios. Additionally, we employ the OntoGraf module within the software to visually display the hierarchical structure and associative relationships of the domain ontology, as depicted in Figure 4. In the figure, solid lines represent the hierarchical relationships of the ontology, while dashed lines illustrate the associations between concept classes.

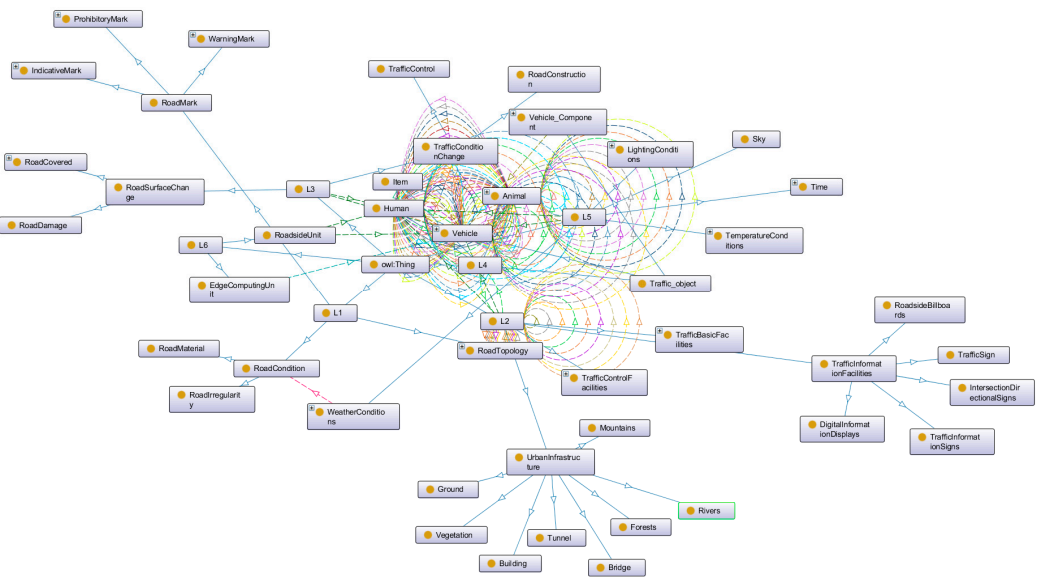


Figure 4. Ontology of driving scenario.

3.3. Knowledge Extraction and Fusion

In the construction of the domain knowledge graph generated in this chapter, the ontology layer mainly relies on manual summarization of human experience. Although it has good organization and theoretical foundation, its scale is limited and difficult to expand in bulk. The data layer, on the other hand, has a large scale and high information richness. However, due to the insufficient organizational structure of the network, the information density of the data layer is relatively low, and its utilization is limited. To address this issue, this section leverages the guiding role of the ontology layer to extract entity data from a large amount of standardized autonomous driving data, serving as the data layer

of the driving scenario domain knowledge graph. By integrating the data layer with the ontology layer, the expansion and correction of the ontology layer are achieved, thereby realizing a "bottom-up" semi-automatic update of the ontology layer.

3.3.1. Data Extraction

Knowledge extraction is the process of extracting and refining knowledge information from data. Depending on the degree of data structure, knowledge extraction can be categorized into structured, semi-structured, and unstructured data extraction. In Section 2.2.3, systematic collection and analysis of open data in the domain have been conducted. After comparative analysis, it was found that the number of relationship labels in the Traffic Genome dataset significantly exceeds that of other datasets. Therefore, the Traffic Genome dataset was selected as the primary target for knowledge extraction.

Programs were written to convert the labeled hdf5 format scene graph into RDF format data. Figure 5 illustrates the conversion of scene graph labels in the Traffic Genome dataset into a small-scale knowledge graph.

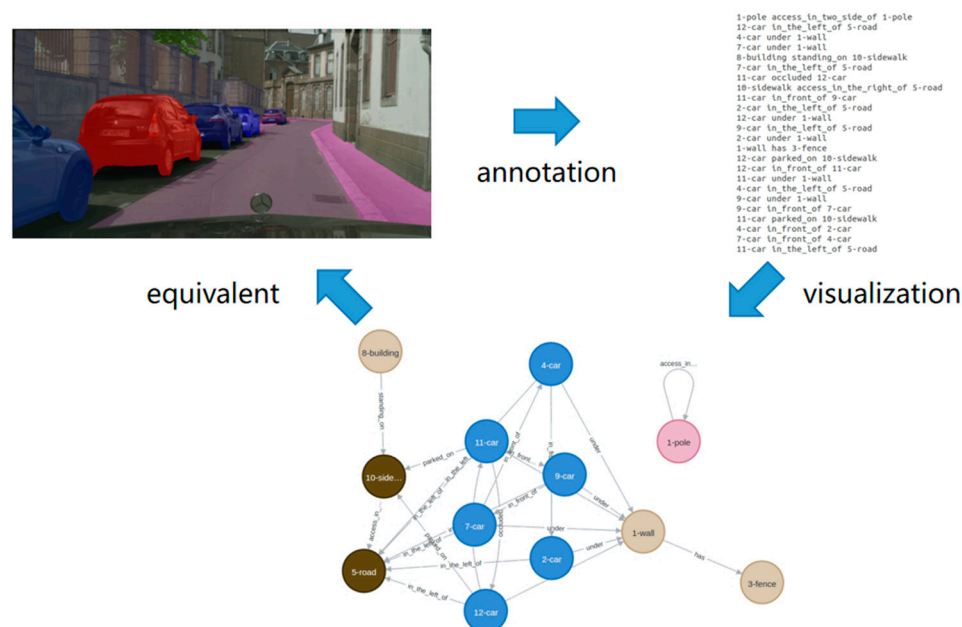


Figure 5. Extraction process of small-scale knowledge graph in the Traffic Genome dataset.

3.3.2. Knowledge Fusion

Knowledge fusion is a process that integrates, disambiguates, processes, verifies, and updates heterogeneous data, information, methods, experiences, and human thoughts into a high-quality knowledge base through high-level knowledge organization under the same framework. Considering that this project involves the fusion between the top-level ontology and a small-scale knowledge graph, a text-based approach can be used for entity matching.

A text-based approach refers to matching entities based on their textual description information. Descriptive information of concepts from two graphs can be extracted, and the similarity between two concepts can be measured by calculating the similarity. For example, in different graphs, the concepts "vehicle" and "car," although having different names, have the same meaning and description, requiring the establishment of a matching relationship between these two concepts. Figure 6 depicts the graphical representation of classes defined in the Traffic Genome dataset, mapped to their equivalent DSKG classes. The equivalence between elements is defined by the "owl:sameAs" relation, indicating that one element can be replaced by another without changing the meaning, and vice versa.

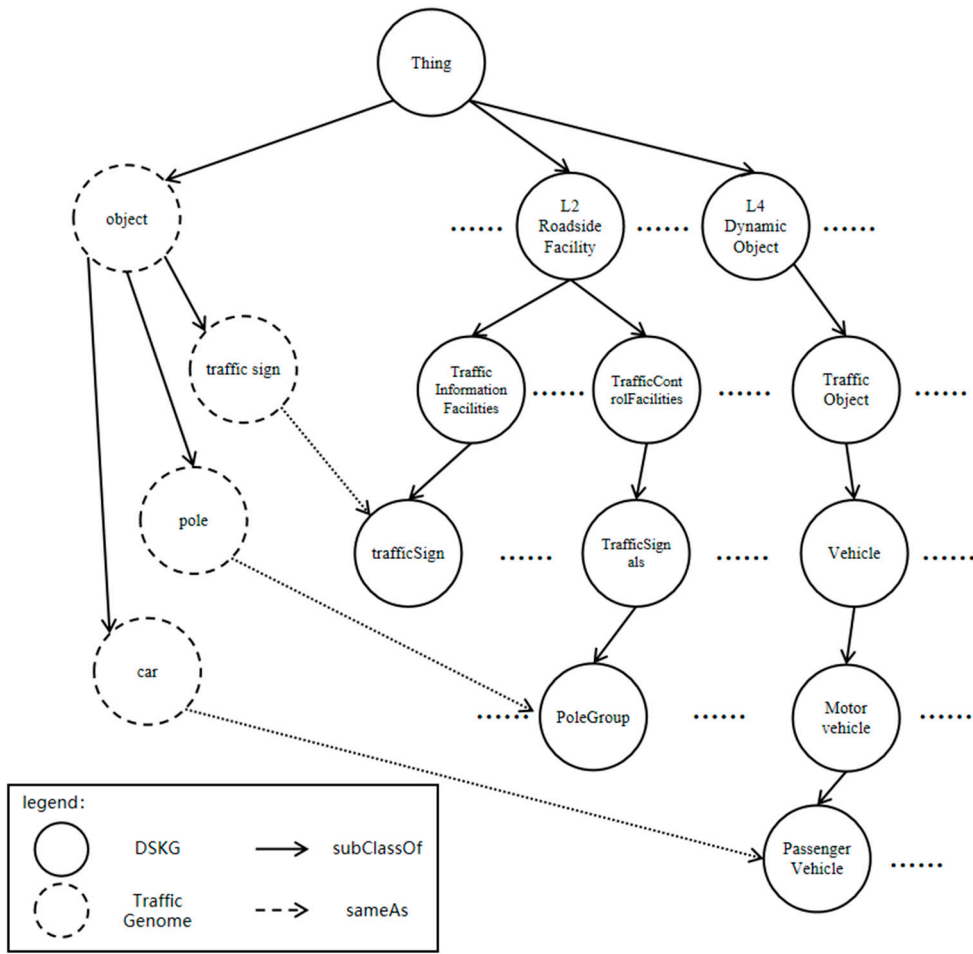


Figure 6. Entity matching diagram.

Through extraction and fusion, the initial knowledge graph of driving scenarios has been constructed. For convenient association, it has been converted and stored in the Neo4j graph database. Neo4j supports large-scale data storage, effectively addressing issues such as low data density, large volume, and rapid updates in the traffic domain. Additionally, the Cypher graph query language supports relevant queries and graph algorithms, facilitating data querying and value mining. Therefore, Neo4j graph database was chosen for knowledge storage in this study. Figure 7 presents the final driving scenario domain knowledge graph (partial):

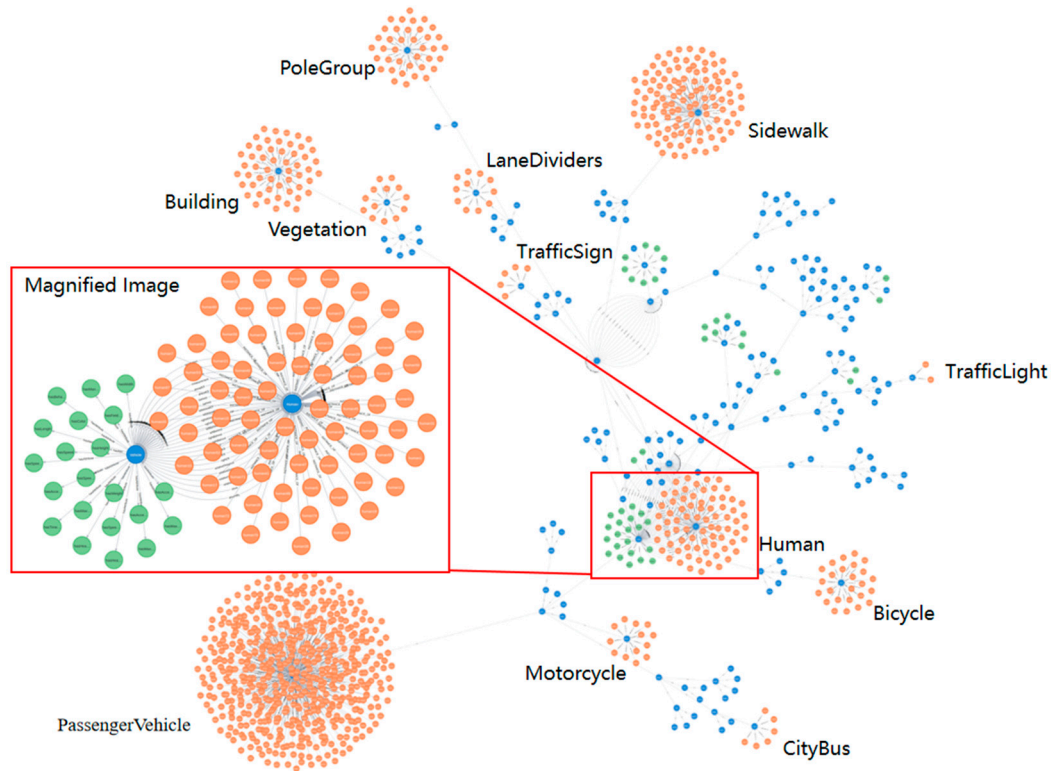


Figure 7. Integrated updated knowledge graph of the driving scenario domain (partial).

3.4. Knowledge Inference Models Based on Representation Learning

The main idea of knowledge inference based on representation learning is to learn the semantic correlations between entities and relations by mapping them into a low-dimensional continuous vector space. Specifically, knowledge embedding models first project the knowledge graph into a low-dimensional vector space, transforming entities and relations in the graph into low-dimensional vectors. Then, a scoring function is designed to compute the scores of all knowledge triples, and the backpropagation algorithm is used to maximize the scores of the triples actually existing in the knowledge graph, thereby learning the vector embeddings of entities and relations in the knowledge graph.

3.4.1. Model Definition

Knowledge graph representation learning models are a class of methods used to learn vector representations of entities and relations in knowledge graphs. These models play a crucial role in entity linking prediction tasks in knowledge graphs. This section introduces several commonly used knowledge graph representation learning models, including the TransE, Complex, Distmult, and RotatE models.

The TransE model is a distance-based knowledge representation learning model. Its core idea is to learn the embedding representation of knowledge by translating the relation vector as the translation operation from the head entity vector to the tail entity vector. For a given triple, where h represents the head entity vector, t represents the tail entity vector, and r represents the relation vector, its scoring function is defined as follows:

$$f_r(h, t) = \|h + r - t\|_2 \quad (1)$$

where $\|\cdot\|_2$ is the norm (usually 1 or 2), and $\|\cdot\|_2$ is the norm. This function calculates the distance between the head entity after relation translation and the tail entity, where a smaller distance indicates a better match.

The Complex model is a complex-based knowledge representation learning model. Its core idea is to represent entities and relations in the knowledge graph as complex vectors and use complex inner product to compute the similarity between entities and relations. Its scoring function is defined as follows:

$$f_r(h,t) = \text{Re}(h^T r \bar{t}) \tag{2}$$

where Re represents the real part, and \bar{t} represents the conjugate of the tail entity vector.

The Dismult model is a dot-product-based knowledge representation learning model. Its core idea is to represent entities and relations in the knowledge graph as vectors and use dot product to measure the similarity between entities and relations. Its scoring function is defined as follows:

$$f_r(h,t) = h^T r \odot t \tag{3}$$

where \odot represents the element-wise product.

The RotatE model is a rotation-based knowledge representation learning model. Its core idea is to use rotation operations to capture semantic correlations between entities and relations. Its scoring function is defined as follows:

$$f_r(h,t) = \text{dist}(h \circ r, t) \tag{4}$$

where \odot represents element-wise multiplication of complex numbers, dist represents a metric function, usually 1 or 2-norm.

3.4.2. Model Training and Analysis

Multiple evaluation metrics were employed in this study to assess the performance of different models in link prediction tasks, including Mean Rank (MR), Mean Reciprocal Rank (MRR), Hits@1, Hits@3, and Hits@10. MR represents the average rank of correct triples in the test set. Hits@10 indicates the ratio of correct triples among the top 10 rankings to the total number of triples in the test set. When the average rank is lower, the MRR, Hits@1, Hits@3, and Hits@10 are higher, indicating better inference performance. The optimal hyperparameters for different models are shown in Table 5.

Table 5. Hyperparameter values of the different models.

Model	Learning Rate	Hidden Layer	Margin	Batch_Size
TransE	0.001	50/256/512	0.9	20000
Complex	0.001	50/256/512	None	20000
Dismult	0.001	50/256/512	0.9	20000
Rotate	0.001	50/256/512	1.0	20000

For a deeper understanding of the model training process, we provide Figure 8, which details the training process. Additionally, Table 6 showcases the entity prediction results in the Traffic Genome dataset. These results offer insights into the accuracy, precision, recall, and other metrics of each model in capturing entities within driving scenarios.

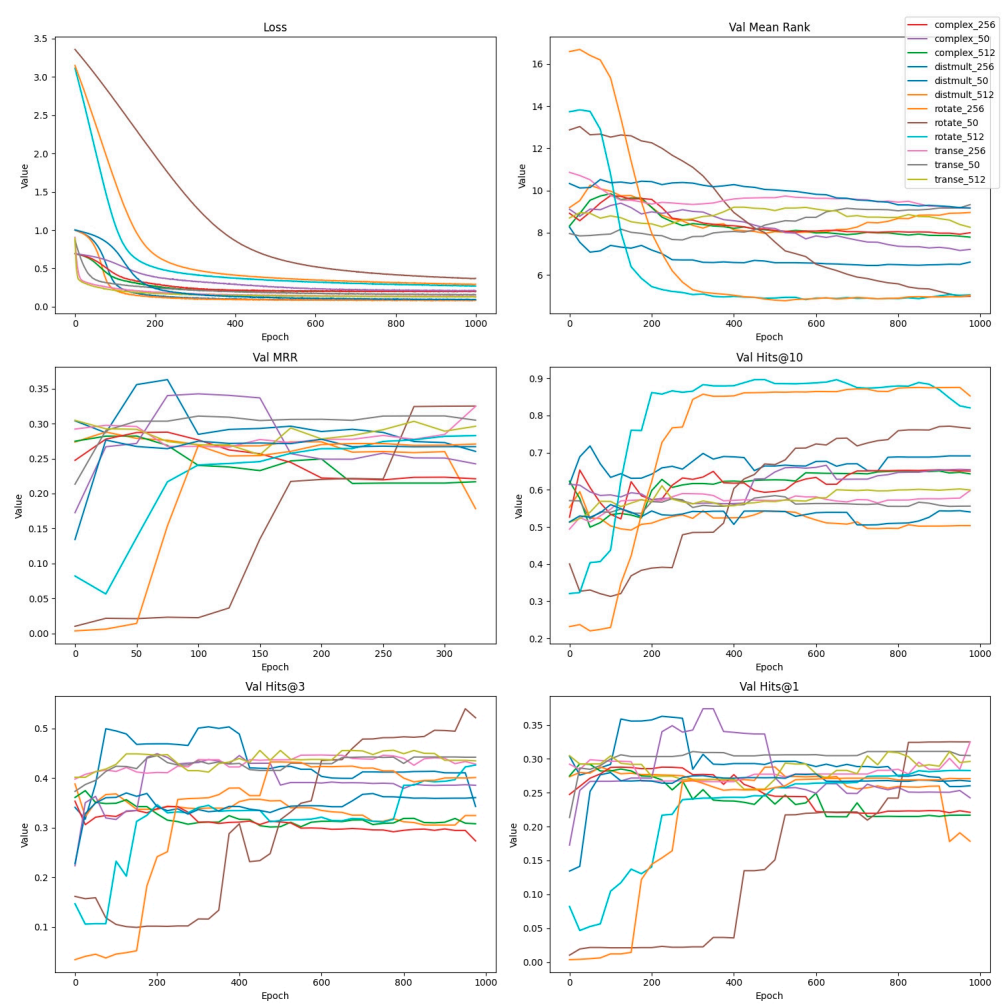


Figure 8. Details of the Model Training Process.

Table 6. Results of predicting the entities.

Model (Optimal Hidden Layer)	MR	MRR	Hits@10	Hits@3	Hits@1
Transe (256)	9.18	41.67%	59.79%	42.82%	32.47%
Complex (50)	8.51	42.01%	56.43%	42.98%	33.77%
Distmult (256)	7.01	46.09%	65.91%	46.92%	36.29%
Rotate (50)	4.99	45.68%	76.54%	52.17%	32.51%

Based on Figure 8 and Table 6, we conducted a comprehensive analysis of four different models. The Rotate model performs best in MR, with an average rank of 4.99, significantly lower than other models. The Complex model excels in MRR, reaching 42.01%, slightly higher than the Rotate model's 45.68%. However, under stricter evaluation metrics such as Hits@10, Hits@3, and Hits@1, the Rotate model demonstrates outstanding performance, ranking first in Hits@10, Hits@3, and Hits@1 with 76.54%, 52.17%, and 32.51% respectively. The Distmult model also performs well, ranking second in Hits@10 with 65.91%, and second in Hits@3 and Hits@1 with 46.92% and 36.29% respectively. In contrast, the Transe and Complex models show relatively average performance across all metrics. Therefore, considering the overall performance, the Rotate model emerges as a preferred choice among these evaluation metrics and could serve as one of the primary models in research.

4. Conclusions

This study investigates the modeling methods of knowledge graphs in the domain of driving scenarios. It includes ontology construction, knowledge extraction and fusion, and analysis of knowledge reasoning models based on representation learning. Ontology construction standardizes domain knowledge. It ensures the wide applicability of concept categories. This process lays the foundation for subsequent knowledge extraction and fusion. In the data extraction and fusion phase, entity data were extracted from the Traffic Genome dataset. It was then integrated with the ontology layer. This facilitated the expansion and updating of the knowledge graph. Finally, in terms of knowledge reasoning models, an analysis of different models' performance in entity linking prediction tasks was conducted. The analysis revealed that the Dismult model exhibited the best performance in metrics such as average ranking and hit rate. The knowledge embedding model based on Dismult provides more accurate scene understanding and decision support for intelligent driving systems.

Author Contributions: Conceptualization, C.Z.; methodology, C.Z.; software, C.Z.; validation, C.Z. and X.L.; formal analysis, D.W.; investigation, L.H.; resources, Y.L.; data curation, J.Y.; writing—original draft preparation, C.Z.; writing—review and editing, D.W. and X.L.; visualization, J.Y.; supervision, L.H.; project administration, Y.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data for this study is publicly available. Below is the link to access the dataset: dskg: <https://github.com/ZhangCeGitHub/dskg/tree/main>

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

References

1. ISO 21448:2022 Road vehicles-safety of the intended functionality. Available online: <https://www.iso.org/obp/ui#iso:std:iso:21448:ed-1:v1:en> (accessed on 28 March 2024).
2. Wickramarachchi R, Henson C, Sheth A. CLUE-AD: a context-based method for labeling unobserved entities in autonomous driving data. *Proceedings of the AAAI Conference on Artificial Intelligence*. **2023**, 37(13), 16491-16493.
3. SINGHAL A. Introducing the knowledge graph: things, not strings. Available online: <https://blog.google/products/search/introducing-knowledge-graph-things-not/> (accessed on 28 March 2024).
4. Geyer S, Baltzer M, Franz B, et al. Concept and development of a unified ontology for generating test and use - case catalogues for assisted and automated vehicle guidance. *IET Intelligent Transport Systems*. **2014**, 8(3), 183-189.
5. Schuldt F, Saust F, Lichte B, et al. Effiziente systematische Testgenerierung für Fahrerassistenzsysteme in virtuellen Umgebungen. *Automatisierungssysteme, Assistenzsysteme und Eingebettete Systeme Für Transportmittel*. **2013**, 4, 1-7
6. Schuldt F. Ein Beitrag für den methodischen Test von automatisierten Fahrfunktionen mit Hilfe von virtuellen Umgebungen. doktors, Technische Universität Braunschweig, Braunschweig, 2017.
7. Audi G, Volkswagen AG, et al. The PEGASUS Method. Available online: <https://www.pegasusprojekt.de/en/pegasus-method> (accessed on 28 March 2024).
8. Bagschik G, Menzel T, Maurer M. Ontology based scene creation for the development of automated vehicles. *IEEE Intelligent Vehicles Symposium*. **2018**, 1813-1820.
9. Bock J, Krajewski R, Eckstein L, et al. Data basis for scenario-based validation of HAD on highways. *Aachen Colloquium Automobile and Engine Technology*. **2018**, 8-10.
10. Bagschik G, Menzel T, Körner C, et al. Wissensbasierte szenariengenerierung für betriebsszenarien auf deutschen autobahnen. *Workshop Fahrerassistenzsysteme und automatisiertes Fahren*. **2018**, 12, 12.

11. Weber H, Bock J, Klimke J, et al. A framework for definition of logical scenarios for safety assurance of automated driving. *Traffic injury prevention*. **2019**, 20: 65-70.
12. Scholtes M, Westhofen L, Turner LR, et al. 6-layer model for a structured description and categorization of urban traffic and environment. *IEEE Access*. **2021**, 9:59131-59147.
13. Report on Advanced Technology Research of Functional Safety for Intelligent Connected Vehicles. Available online: <http://www.caicv.org.cn/index.php/material?cid=38> (accessed on 28 March 2024).
14. Bagschik G, Menzel T, Maurer M. Ontology based scene creation for the development of automated vehicles. *IEEE Intelligent Vehicles Symposium*. **2018**, 1813-1820.
15. Menzel T, Bagschik G, Isensee L, et al. From functional to logical scenarios: Detailing a keyword-based scenario description for execution in a simulation environment. *IEEE Intelligent Vehicles Symposium*. **2019**, 2383-2390.
16. Tahir Z, Alexander R. Intersection focused situation coverage-based verification and validation framework for autonomous vehicles implemented in Carla. *International Conference on Modelling and Simulation for Autonomous Systems*. **2021**, 191-212.
17. Herrmann M, Witt C, Lake L, et al. Using ontologies for dataset engineering in automotive AI applications. *Design, Automation & Test in Europe Conference & Exhibition*. **2022**, 526-531.
18. ASAM OpenXOntology. Available online: <https://www.asam.net/standards/asam-openxontology/> (accessed on 28 March 2024).
19. Bogdoll D, Guneshka S, Zöllner JM. One ontology to rule them all: Corner case scenarios for autonomous driving. *European Conference on Computer Vision*. **2022**, 409-425.
20. Westhofen L, Neurohr C, Butz M, et al. Using ontologies for the formalization and recognition of criticality for automated driving. *IEEE Open Journal of Intelligent Transportation Systems*. **2022**, (3), 519-538.
21. Geiger A, Lenz P, Stiller C, et al. Vision meets robotics: The kitti dataset. *The International Journal of Robotics Research*. **2013**, 32(11), 1231-1237.
22. Yu F, Chen H, Wang X, et al. Bdd100k: A diverse driving dataset for heterogeneous multitask learning. *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. **2020**, 2636-2645.
23. Caesar H, Bankiti V, Lang AH, et al. nuscenes: A multimodal dataset for autonomous driving. *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. **2020**, 11621-11631.
24. Sun P, Kretschmar H, Dotiwalla X, et al. Scalability in perception for autonomous driving: Waymo open dataset. *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. **2020**, 2446-2454.
25. Geyer J, Kassahun Y, Mahmudi M, et al. A2d2: Audi autonomous driving dataset. *arXiv preprint*. **2020**, 2004.06320.
26. Xiao P, Shao Z, Hao S, et al. Pandaset: Advanced sensor suite dataset for autonomous driving. *IEEE International Intelligent Transportation Systems Conference*. **2021**, 3095-3101.
27. Zhang Z, Zhang C, Niu Z, et al. Geneannotator: A semi-automatic annotation tool for visual scene graph. *arXiv preprint*, **2021**, 2109.02226.
28. Cordts M, Omran M, Ramos S, et al. The cityscapes dataset for semantic urban scene understanding. *Proceedings of the IEEE conference on computer vision and pattern recognition*. **2016**, 3213-3223.
29. Jia M, Zhang Y, Pan T, et al. Ontology Modeling of Marine Environmental Disaster Chain for Internet Information Extraction: A Case Study on Typhoon Disaster. *J. Geo-Inf. Sci.* **2020**, 22, 2289-303.
30. Ghosh SS, Chatterjee SK. A knowledge organization framework for influencing tourism-centered place-making. *Journal of Documentation*. **2022**, 78(2), 157-176.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.