豊田工業大学
TOYOTA TECHNOLOGICAL INSTITUTE

知能数理研究室
Computational Intelligence Lab

Semantic Technology and <u>Level 5</u> Autonomous Driving

Computational Intelligence (COIN) Laboratory
Toyota Technological Institute

November 28, 2018

Yutaka Sasaki

Toyota Technological Institute (TTI)

- Established by Toyota Motor Corp. in 1981 as one of its social contributions.
- Compact university admitting about 100 undergrads and about 45 postgrads each year.
- Located in Nagoya City.

(Disclaimer: This presentation does not express any technical standpoint of Toyota companies.)

Toyota Technological Institute (cont.)

- One faculty with 3 courses:
  - mechanical system engineering
  - electronics & information engineering
  - material science and engineering.
- 46 faculty members closely related.
- Established Toyota Technological Institute at Chicago (TTIC) on the University of the Chicago campus in 2003.
  - KITTI (vision data set)

TTI Research Center for Smart Vehicles

- Established in 2010.
- Research goals:
  - Develop key technologies essential for realizing Smart Vehicles,
  - Targeting cutting-edge technologies suitable for university laboratories.

NB: “Smart Vehicle” means any kinds of intelligent unmanned vehicles, including autonomous robots and UAVs.

Self-driving Technology

SAE levels of automated driving

SAE level	Name	Narrative Definition	Execution of Steering and Acceleration/ Deceleration	*Monitoring* of Driving Environment	Fallback Performance of *Dynamic Driving Task*	System Capability *(Driving Modes)*
***Human driver* monitors the driving environment**						
0	**No Automation**	the full-time performance by the *human driver* of all aspects of the *dynamic driving task*, even when enhanced by warning or intervention systems	Human driver	Human driver	Human driver	n/a
1	**Driver Assistance**	the *driving mode*-specific execution by a driver assistance system of either steering or acceleration/deceleration using information about the driving environment and with the expectation that the *human driver* perform all remaining aspects of the *dynamic driving task*	Human driver and system	Human driver	Human driver	Some driving modes
2	**Partial Automation**	the *driving mode*-specific execution by one or more driver assistance systems of both steering and acceleration/ deceleration using information about the driving environment and with the expectation that the *human driver* perform all remaining aspects of the *dynamic driving task*	**System**	Human driver	Human driver	Some driving modes
***Automated driving system* ("system") monitors the driving environment**						
3	**Conditional Automation**	the *driving mode*-specific performance by an *automated driving system* of all aspects of the dynamic driving task with the expectation that the *human driver* will respond appropriately to a *request to intervene*	System	**System**	Human driver	Some driving modes
4	**High Automation**	the *driving mode*-specific performance by an automated driving system of all aspects of the *dynamic driving task*, even if a *human driver* does not respond appropriately to a *request to intervene*	System	System	**System**	Some driving modes
5	**Full Automation**	the full-time performance by an *automated driving system* of all aspects of the *dynamic driving task* under all roadway and environmental conditions that can be managed by a *human driver*	System	System	System	**All driving modes**

(https://www.sae.org/misc/pdfs/automated_driving.pdf)

Research areas of self-driving

Hardware

Sensors:
  Lidar, millimeter radar, sonar,
  GPS, steering angle, velocity,
  accelerometer,
  camera (stereo, IR, FIR)
Body:
  motor, steering, brake, accelerator,
  body material, shape, ...

Software

Object/position detection:
  lane, car, human, bicycle, ...
Scene understanding：
  object movement, intension,
  collision detection
Control:
  steering, speed, path planning
High-level control:
  knowledge base, dialogue,
  reasoning

Infrastructure

Intelligent Transport System (ITS):
  vehicle-to-vehicle communication,
  driving safety support
    (digital traffic signal, traffic regulation info, road side sensor),
  3D digital road map

Why we need driving knowledge bases?

- In the fully automated (level 5) driving,
  – cars should do (without a human driver):
    - Careful consideration of passengers/destinations
    - Decision making in some exceptional road conditions
    - Conversations, like with a taxi driver
    - In accidents, the car should know what to do
    - Avoidance of criminal usages

⇩

Almost equivalent to creating AI robots!

Comparison with autonomous robots

- In a general sense, autonomous cars are a kind of autonomous robots, sometimes called as robot cars.
- Autonomous cars are constrained by:
  - Strict regulations such as traffic laws.
  - Underactuated manipulation: Inputs are steering and brake/accelerator (gas) for 6 degrees of freedom.
  - Highest priority in safe driving.

For knowledge-based driving, these constraints are not disadvantages but can be advantages.

Three Laws of Robotics

1. A robot may not injure a human being or, through inaction, allow a human being to come to harm.
2. A robot must obey the orders given it by human beings except where such orders would conflict with the First Law.
3. A robot must protect its own existence as long as such protection does not conflict with the First or Second Laws.

Isaac Asimov, "Runaround“, I, Robot, 1950.

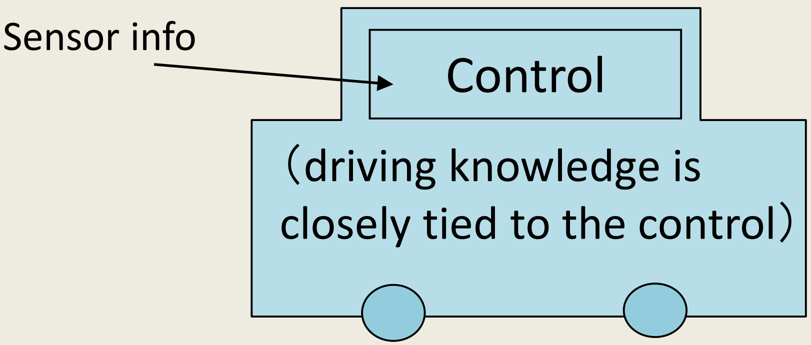
Three Laws of Autonomous Vehicles

1. An autonomous vehicle may not injure a human being or, through inaction, allow a human being to come to harm.
2. An autonomous vehicle must obey laws except where such orders would conflict with the First Law.
3. An autonomous vehicle must obey the orders given it by human beings except where such orders would conflict with the First or Second Laws.

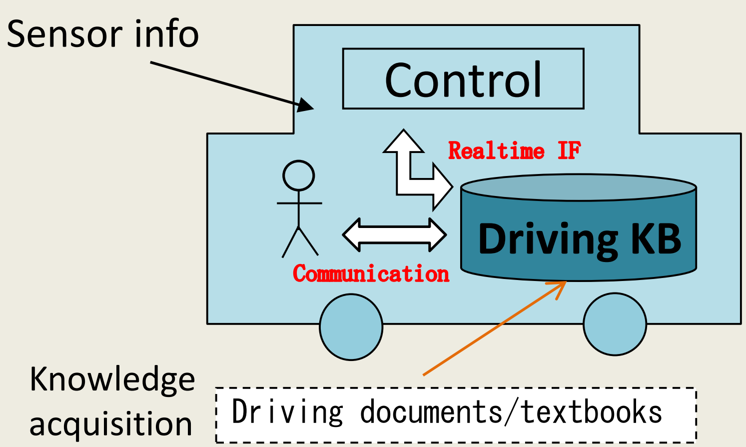
Yutaka Sasaki, presentation material, Smart Vehicles Research Center Symposium 2016. (in Japanese)

Our approach to level 5 self-driving

Current self-driving



Introduce a KB



Research topics:
- How to model driving knowledge bases?
- How to acquire driving knowledge?
- How to manage the knowledge bases?
- How to achieve real time performance?
- How to evaluate KB quality?

Advantages of driving knowledge bases

- Enable:
  - to obey traffic laws while driving.
  - to follow driving manners, e.g. hazard flashers.
  - (shallow) reasoning like human drivers do.
  - to make conversation, like with a taxi driver.
  - to drive carefully for safety, like slowing down when seeing some children on a side walk.
  - to make decisions using common sense, e.g. inoperative traffic signals.
- Can be updated/corrected independent of driving control software.
  - Tailoring a KB for each country/region.

Semantic Technology for self-driving

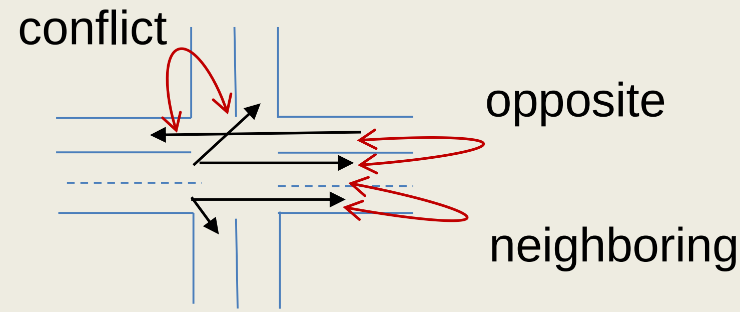
Ontology-based driving (at intersections)

- Ralf Regele, Using Ontology-Based Traffic Models for More Efficient Decision Making of Autonomous Vehicles, Fourth International Conference on Autonomic and Autonomous Systems, 2008.
- B. Hummel, Description logic for scene understanding at the example of urban road intersections, Fakultät für Maschinenbau Universitat Karlsruhe (TH), 2009.
- Michael Hülsen et al., Traffic intersection situation description ontology for advanced driver assistance, IEEE Intelligent Vehicles Symposium (IV), 2011.
- Edson Prestes et al., Core Ontology for Robotics and Automation, Workshop on Standardized Knowledge Representation and Ontologies for Robotics and Automation, 2014.
- Armand et al., Ontology-based context awareness for driving assistance systems, IEEE Intelligent Vehicles Symposium (IV), 2014.
- Brunner et al., 2017 IEEE International Conference on Vehicular Electronics and Safety (ICVES), Ontologies Used in Robotics: A Survey with an Outlook for Automated Driving, 2017
- Lihua Zhao, Ryutaro Ichise, Seiichi Mita, Yutaka Sasaki, An Ontology-Based Intelligent Speed Adaptation System for Autonomous Cars, The 4th Joint International Semantic Technology Conference (JIST2014), 2014
- Lihua Zhao, Ryutaro Ichise, Tatsuya Yoshikawa, Takeshi Naito, Toshiaki Kakinani, Yutaka Sasaki, Ontology-based Decision Making on Uncontrolled Intersections and Narrow Roads, 2015 IEEE Intelligent Vehicles Symposium, Seoul, 2015.
- Lihua Zhao, Ryutaro Ichise, Yutaka Sasaki, Zheng Liu, Tatsuya Yoshikawa, Fast Decision Making using Ontology-based Knowledge Base, 2016 IEEE Intelligent Vehicles Symposium, Gothenburg, Sweden, June 2016.
- Lihua Zhao, Ryutaro Ichise Zheng Liu, Seiichi Mita, and Yutaka Sasaki, Ontology-based Driving Decision Making: A Feasibility Study at Uncontrolled Intersections, IEICE Transactions on Information and Systems, Vol. E100.D, No. 7, pp. 1425-1439, 2017.

Ontology-based Traffic Models

[Regele, 2008]

- CyberCars-2 Project (FP5)
  - "development of a fleet of fully autonomous cybercars which can be used for passenger transport."
  - introduction of road connection networks with conflict, neighboring and opposing relations.



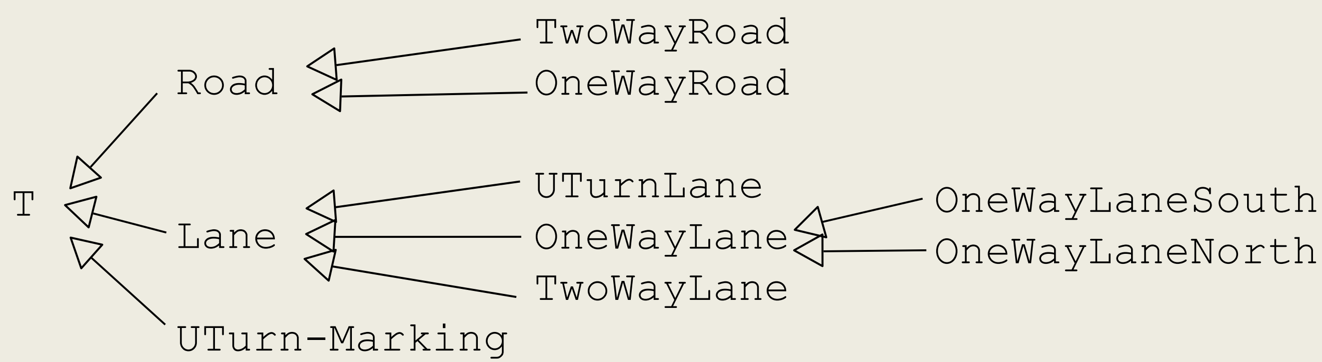
- The model does not contain semantic information:
  - speed limitation, actual geometric information of the roads.

Logical representation of road intersections

[Hummel, 2009]

Formal representations with Description Logic

- Terminological box (TBox)



- Assertion box (ABox)

$\alpha = \{georgeSt: \exists hasPart.(OneWayLaneSouth \sqcup TwoWayLane)\}$

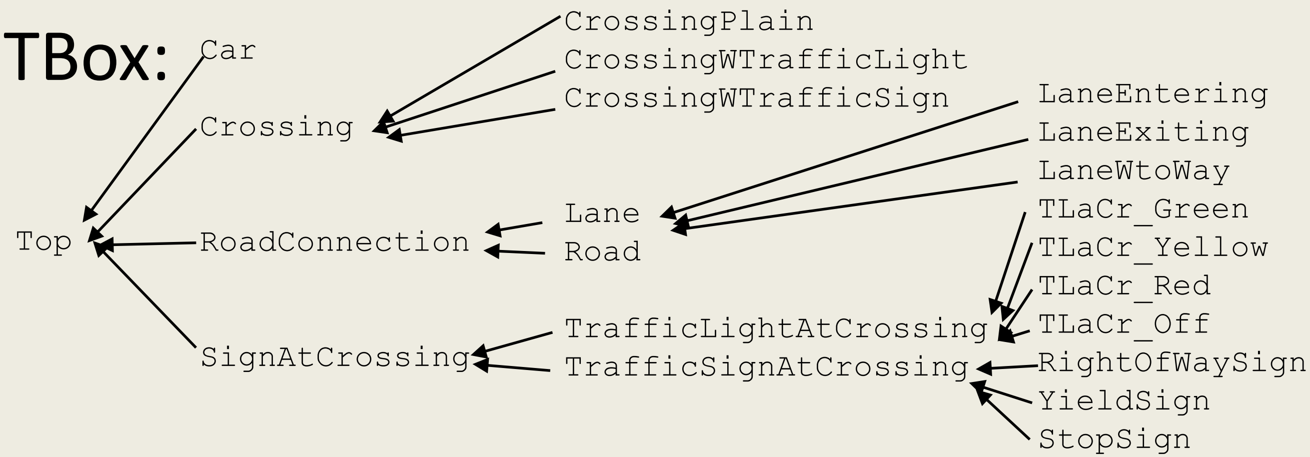
(George St. has some lane with driving direction southwards.)

Traffic Intersection Situation Description Ontology

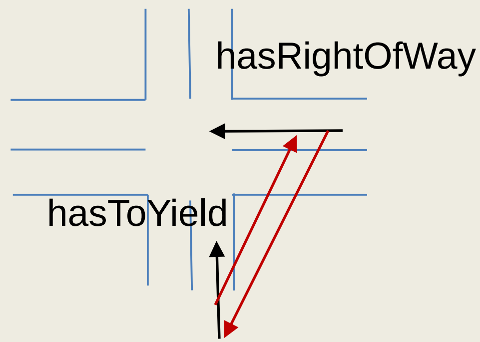
[Hülsen et al. 2011]

- Intersection ontologies contain concepts of car, crossing, road connection (lane and road), and sign at crossing (traffic light and traffic sign).

TBox:



ABox:



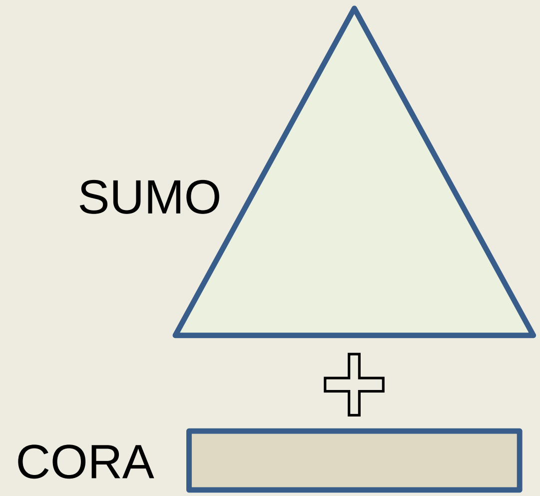
Rule: $Crossing(?cr) \wedge neg\big(CrossingWTrafficLight(?cr)\big) \wedge neg\big(CrossingWTrafficSign(?cr)\big) \rightarrow CrossingPlain(?cr)$

- In a simple intersection case, it takes less than 0.5s for DL reasoning. In a complex case, it takes 1.1-3.6s.
- Not tested with simulation or real-world data.

Core Ontology for Robotics and Automation (CORA)

[Prestes et al. 2014]

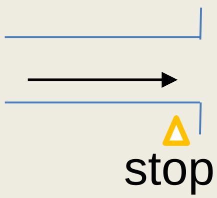
- Drafted by IEEE Standard Association (IEEE SA)
- Extended the Suggested Upper Merged Ontology (SUMO).



Ontology-Based Context Awareness

[Armand et al. 2014]

- An ontology includes context concepts:
  - Mobile Entity (Pedestrian and Vehicle), Static Entity (Road Infrastructure and Road Intersection), context parameters (isClose, isFollowing, and isToReach)
- to understand the context information when it approaches road intersections.
- 14 SWRL rules of spatio-temporal relationships



(example)

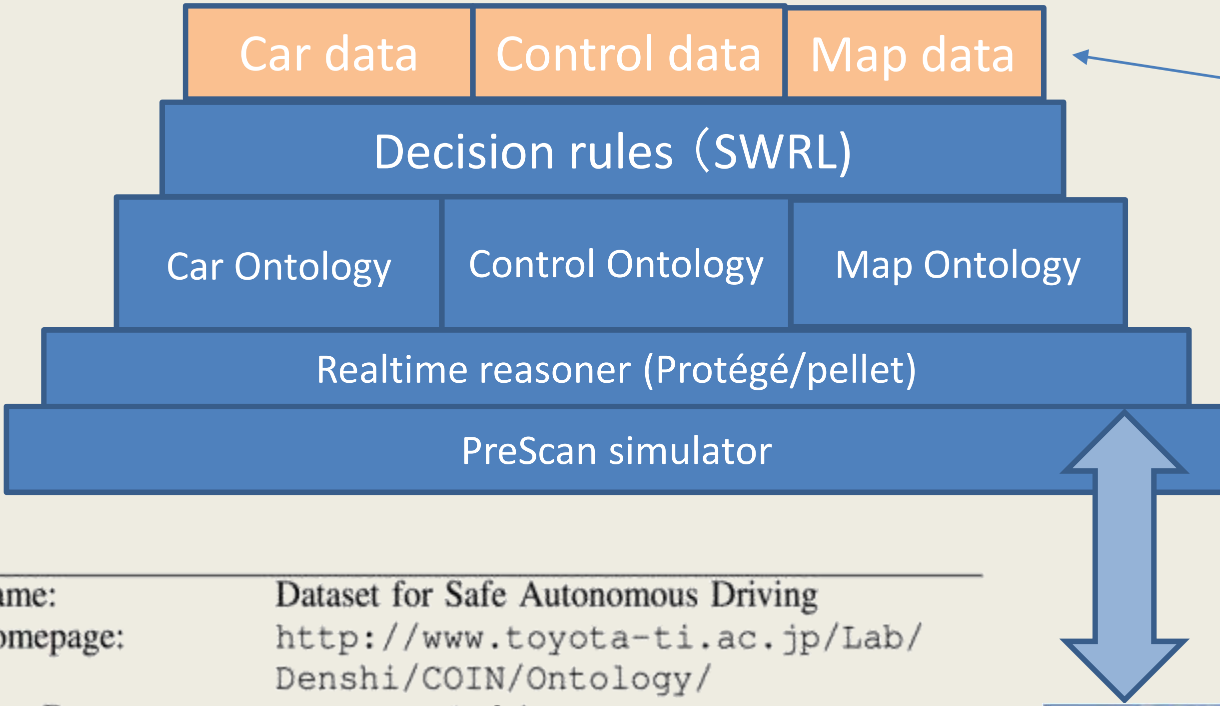
```
vehicle(?v1) ∧ StopIntersection(?stop1) ∧ isToReach(?v1,?stop1)
    → hasToStop(?v1,?stop1)
```

(A vehicle that is about to reach a stop intersection has to stop at the intersection.)

TTI Ontology

[Zhao et al. 2014]

- Driving ontology development since 2009.



Name:	Dataset for Safe Autonomous Driving
Homepage:	http://www.toyota-ti.ac.jp/Lab/Denshi/COIN/Ontology/
Data Dump:	TTICore-0.3/
VoID Description:	void.ttl
RDF Triples:	37566
Entities:	1424
Classes	149
Data Properties	40
Object Properties	35

Experimental vehicle

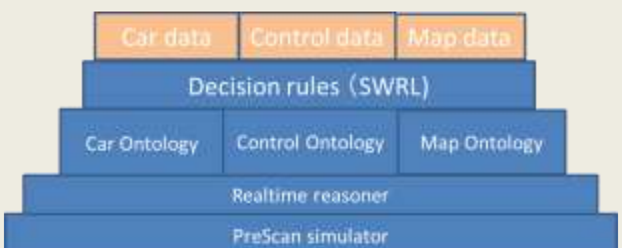
TTI Core Ontology

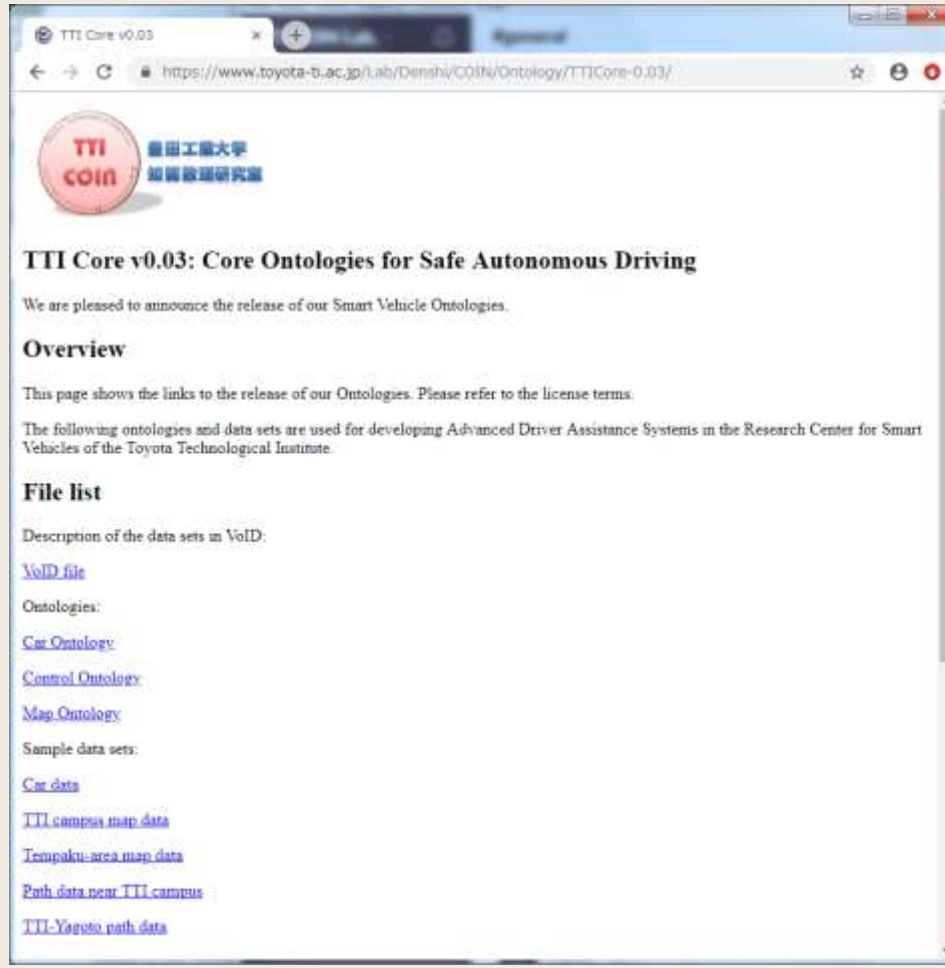
https://www.toyota-ti.ac.jp/Lab/Denshi/COIN/Ontology/TTICore-0.03/

TTI Core v0.03

Ontology
- Car
- Control
- Map

Data
- Car
- Control
- Map





TTI Core v0.03: Core Ontologies for Safe Autonomous Driving

We are pleased to announce the release of our Smart Vehicle Ontologies.

Overview

This page shows the links to the release of our Ontologies. Please refer to the license terms.

The following ontologies and data sets are used for developing Advanced Driver Assistance Systems in the Research Center for Smart Vehicles of the Toyota Technological Institute.

File list

Description of the data sets in VoID:

VoID file

Ontologies:

Car Ontology

Control Ontology

Map Ontology

Sample data sets:

Car data

TTI campus map data

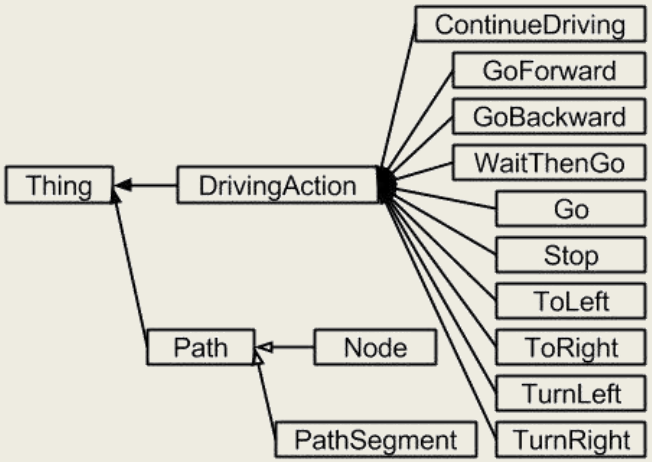
Tempaku-area map data

Path data near TTI campus

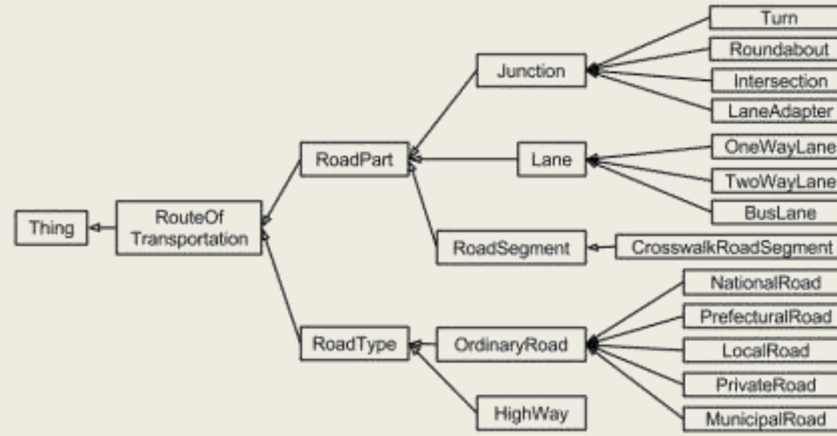
TTI-Yagoto path data

TTI Core: Classes

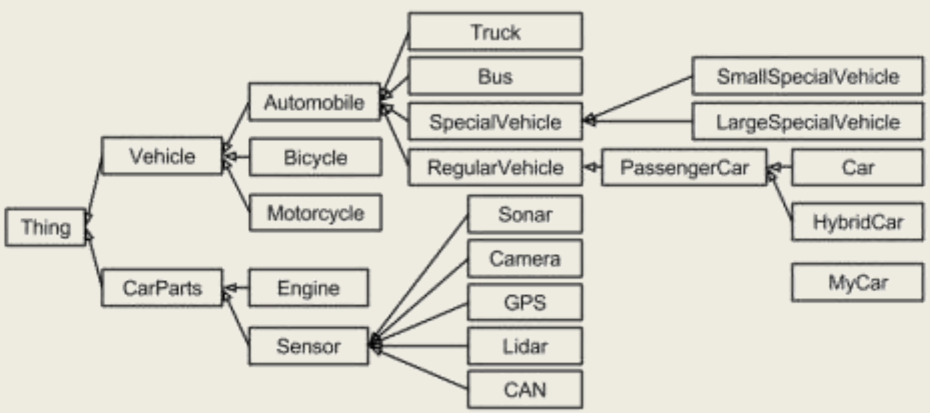
Control class



Map class



Car class



TTI Core: Object/data properties

Control property

- topObjectProperty
  - approachTo
  - collisionWarningWith
  - endLane
  - giveWay
  - hasNode
  - hasPathSegment
  - isNodeOf
    - enterOf
    - exitOf
  - isPathSegmentOf
  - nextNode
  - nextPathSegment
  - prevNode
  - prevPathSegment
  - startLane

- topDataProperty
  - nodePos
  - pathSegmentID

Car property

- topObjectProperty
  - currentPath
  - isRunningOn
  - usedSensor

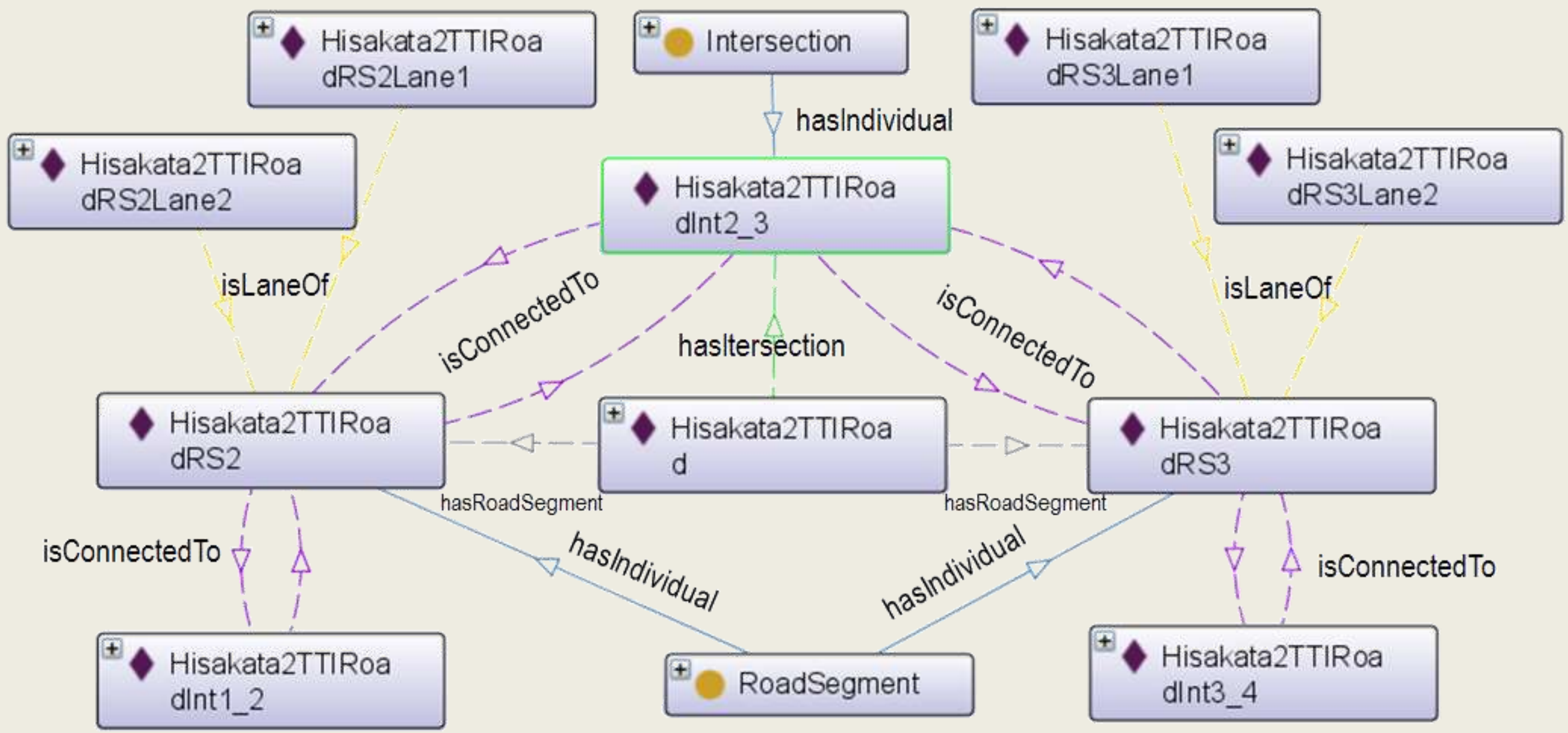
- topDataProperty
  - brake
  - car_height
  - car_length
  - car_model
  - carID
  - distance_back
  - distance_front
  - front_overhang
  - gear
  - height_max
  - rear_overhang
  - steer_angle
  - throttle
  - velocity
  - wheel_base

Map property

- topObjectProperty
  - goStraightTo
  - hasCar
  - hasJunction
    - hasIntersection
  - hasLane
  - hasRoadSegment
  - isConnectedTo
  - isLaneOf
  - isLeftOf
  - isNextTo
  - isOn
  - isRightOf
  - isRoadSegmentOf
  - nearTo
  - relatedTrafficLight
  - turnLeftTo
  - turnRightTo

- topDataProperty
  - address
  - adminArea
  - allowedDir
  - allowLaneChange
  - alt
  - angle
  - boundCoordinate
  - boundPos
  - brand
  - curve_id
  - drivingDir
  - enterPos
  - exitPos
  - isEmpty
  - lat
  - loc_x
  - loc_y
  - loc_z
  - lon
  - name
  - opening_hours
  - operator
  - orientation
  - osm_node_id
  - osm_ref
  - osm_way_id
  - phone
  - postcode
  - pubFacAdmin
  - river_id
  - speedMax
  - website

Visualization of road connections



Instance

Table 1: An example of map ontology based instances.

Subject	Property	Object
tempaku:Hisakata2Road	rdf:type	map:LocalRoad
tempaku:Hisakata2Road	map:speedMax	"40"^^kmh
tempaku:Hisakata2Road	map:osm_ref	osm_way:49559442
tempaku:Hisakata2Road	map:hasIntersection	tempaku:Hisakata2Int5_6
tempaku:Hisakata2Road	map:hasRoadSegment	tempaku:Hisakata2TTIRoadRS1
tempaku:Hisakata2Int5_6	rdf:type	map:Intersection
tempaku:Hisakata2Int5_6	map:boundPos	35.107663, 136.983845
tempaku:Hisakata2Int5_6	map:boundPos	35.107846, 136.983889
tempaku:Hisakata2Int5_6	map:boundPos	35.107860, 136.983683
tempaku:Hisakata2Int5_6	map:boundPos	35.107708, 136.983604
tempaku:Hisakata2Int5_6	map:isConnectedTo	tempaku:Hisakata2RS5
tempaku:Hisakata2Int5_6	map:isConnectedTo	tempaku:Hisakata2TTIRoadRS1
tempaku:Hisakata2RS5	rdf:type	map:RoadSegment
tempaku:Hisakata2RS5	map:boundPos	35.107663, 136.983845
tempaku:Hisakata2RS5	map:boundPos	35.107357, 136.984067
tempaku:Hisakata2RS5	map:isConnectedTo	tempaku:Hisakata2Int5_6
tempaku:Hisakata2RS5Lane1	rdf:type	map:OneWayLane
tempaku:Hisakata2RS5Lane1	map:enterPos	35.107353, 136.984054
tempaku:Hisakata2RS5Lane1	map:exitPos	35.107657, 136.983832
tempaku:Hisakata2RS5Lane1	map:isLaneOf	tempaku:Hisakata2RS5
tempaku:Hisakata2RS5Lane2	rdf:type	map:OneWayLane
tempaku:Hisakata2RS5Lane2	map:enterPos	35.107667, 136.983856
tempaku:Hisakata2RS5Lane2	map:exitPos	35.107362, 136.984081
tempaku:Hisakata2RS5Lane2	map:isLaneOf	tempaku:Hisakata2RS5

speed limit

GPS position

Simulator/real-world Experiments

Table 5: Query the speed limit of currentPathSegment.

```
SELECT ?max
WHERE {
{ tempaku:currentPathSegment map:isLaneOf          ?roadsegment.
  ?roadsegment               map:isRoadSegmentOf   ?road.
  ?road                      map:speedMax          ?max.
} UNION {
  ?road                      map:hasIntersection   map:currentPathSegment.
  ?road                      map:speedMax          ?max. }
}
```

Table 6: C-SPARQL query for checking if a car overspeeds its own speed limit.

```
REGISTER QUERY OverSpeedCheck AS
SELECT ?car
FROM STREAM <http://www.toyota-ti.ac.jp/Lab/Denshi/COIN/stream>
  [RANGE 500ms STEP 50ms]
WHERE { ?car car:velocity ?speed . }
GROUP BY ?speed
HAVING (AVG(?speed) >= maxSpeed )
```

(a) PreScan Simulator Car. (b) Intelligent Car.

Fig. 6: PreScan simulator car and intelligent car.

(a) PreScan Trajectory. (b) Intelligent Car Trajectory.

Fig. 7: Trajectory for the experiment.

Table 7: Instances in the Knowledge Base.

Types of Instances	Number of Instances	Speed Limit
Intersection	54	
RoadSegment	59	
Kindergarten	1	
BusLane	5	
OneWayLane	162	
PrivateRoad	1	25km/h
MunicipalRoad	1	50km/h
LocalRoad	5	40km/h (3), 50km/h (2)
Path	1	

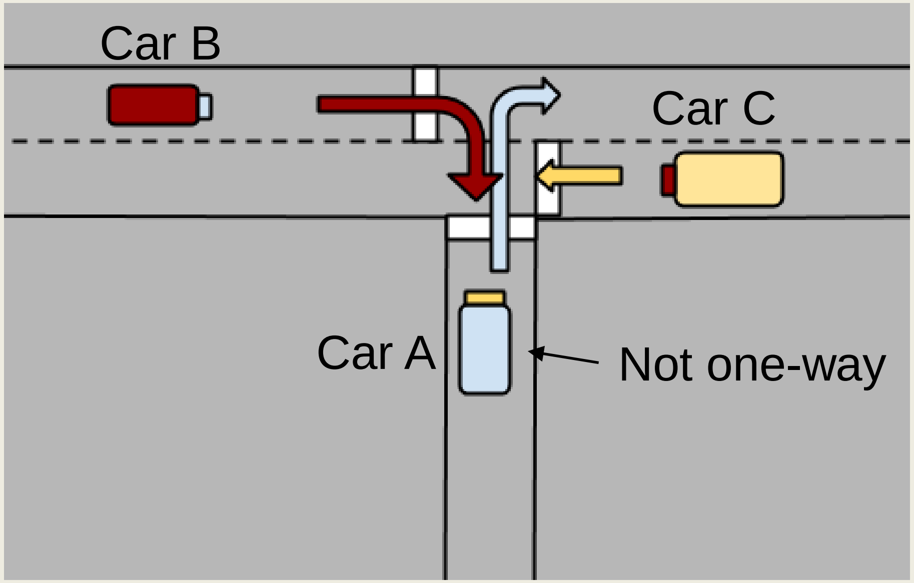
- Simulation:
  Reasoning time: 242ms
  Query time: 2-23ms
- Real-world (Off line):
  Reasoning: 117ms
  Query time: 3-23ms

Ontology-based decision making at uncontrolled intersections

Targeted traffic scenario

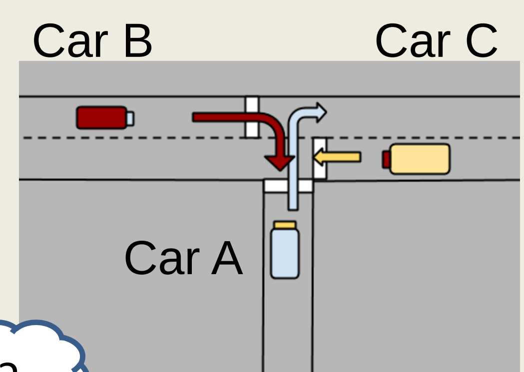
[Zhao et al. 2017]

- Real world problems at uncontrolled intersections, i.e. no traffic lights.
- In Japan, there are a lot of narrow roads where even human drives feel difficulty in driving.



Issues

- Car A has the lowest priority because it cannot interfere traffics on the wide road bylaw.
- Car B has higher priority than Car A.
- Car C has the highest priority.
- The narrow road is difficult for two vehicles to run freely; It is safer to stop on the left side and give way to the other vehicle to pass by slowly.
- In this case, Car A should go out of the narrow road before Car B for safe driving.

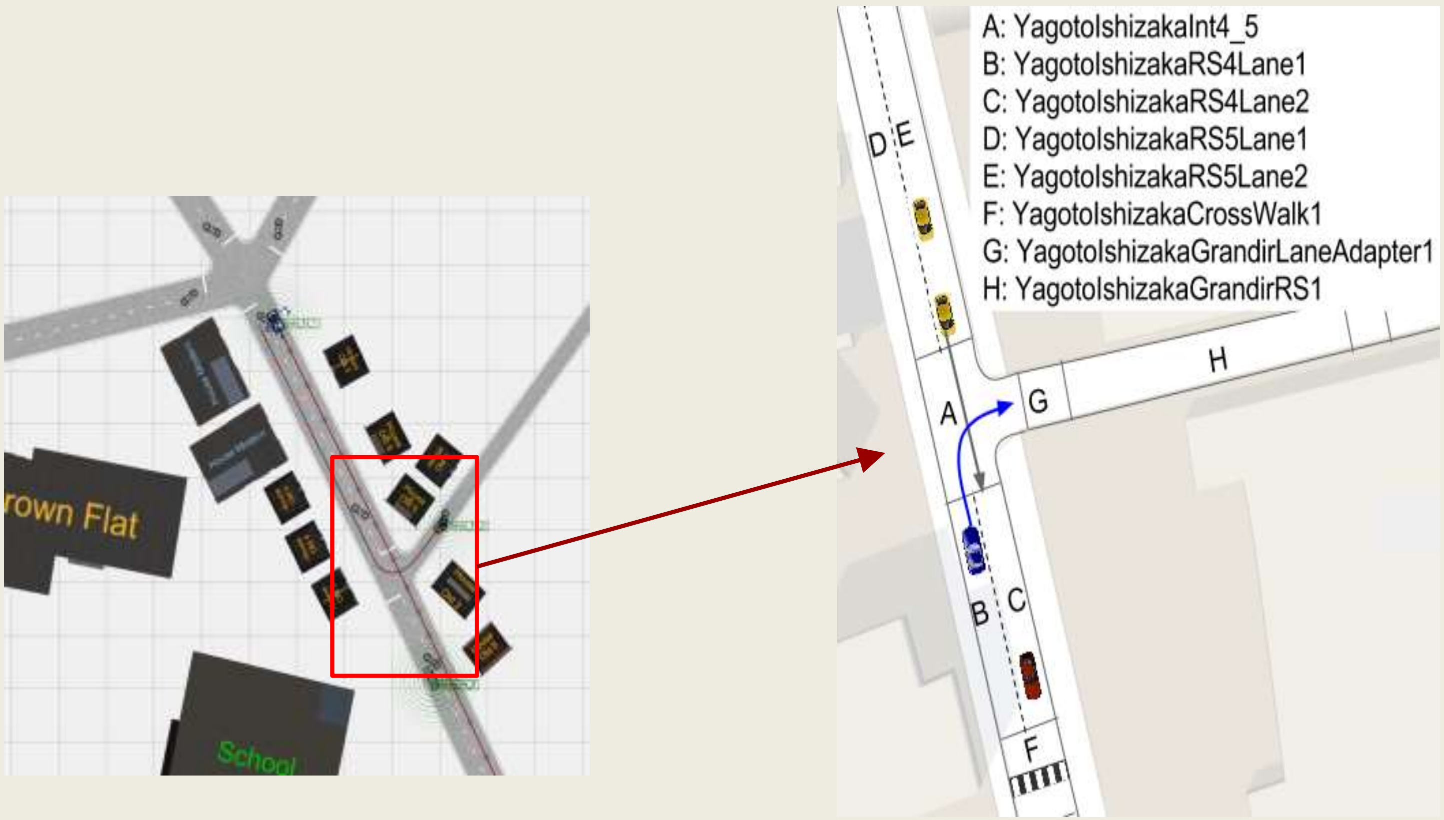


Level 5 autonomous cars should solve this situation with some reasoning.

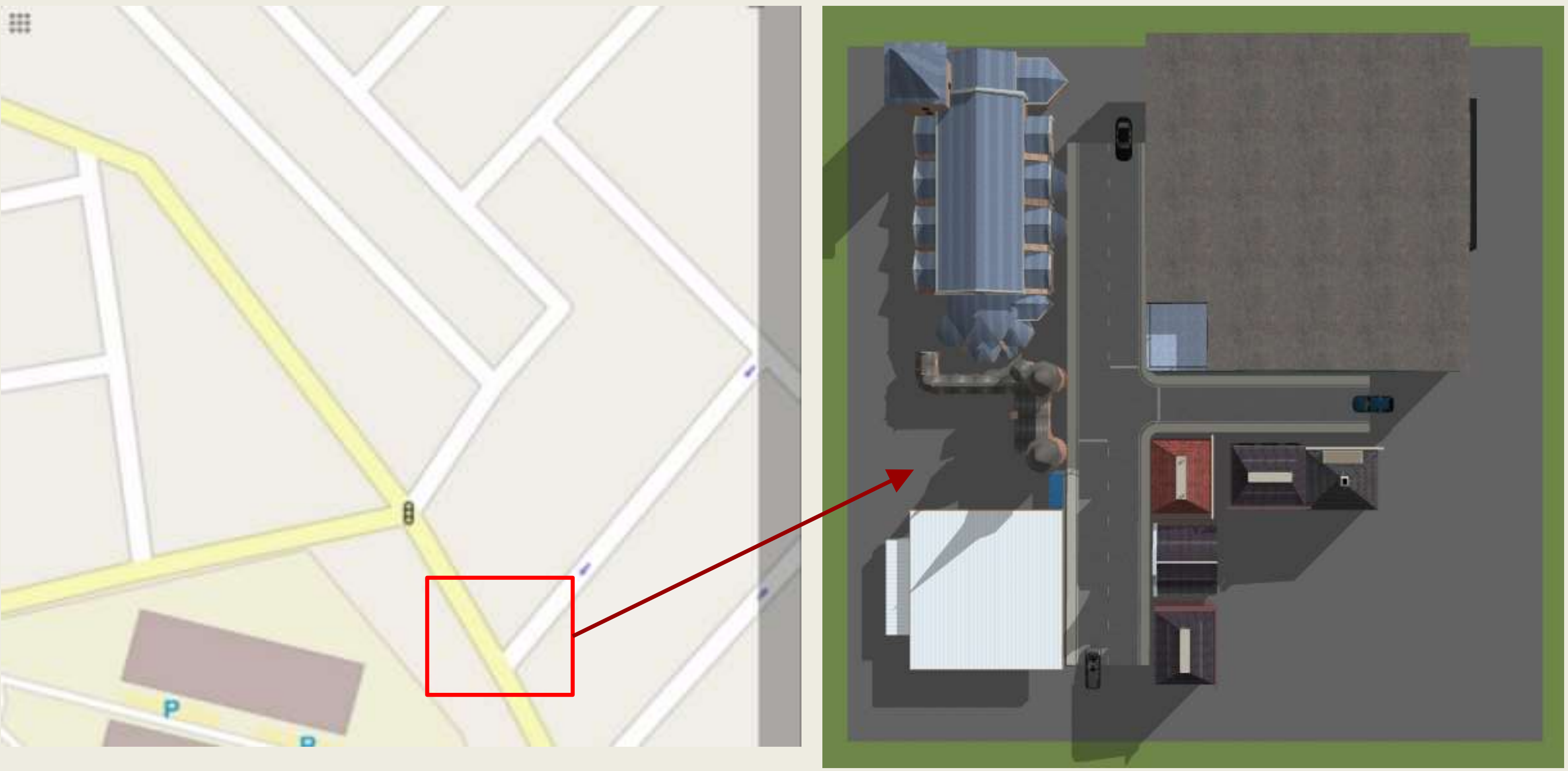
How?

Map data creation

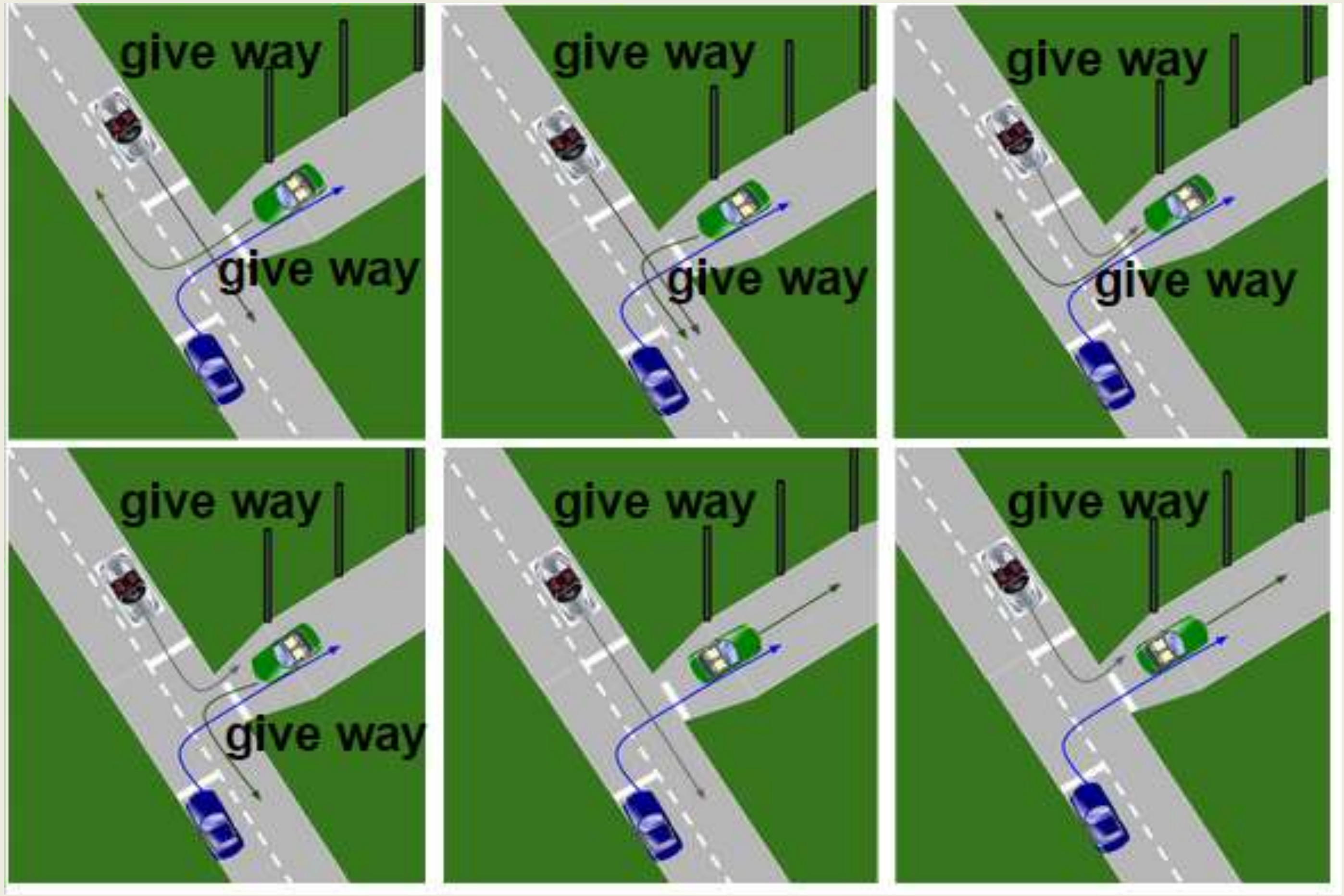
- Using OpenStreetMap data, we created a map data set for our ontology.
- Lane connections are manually considered.
- Independent of map data used for the drive control.



Created 3D similation

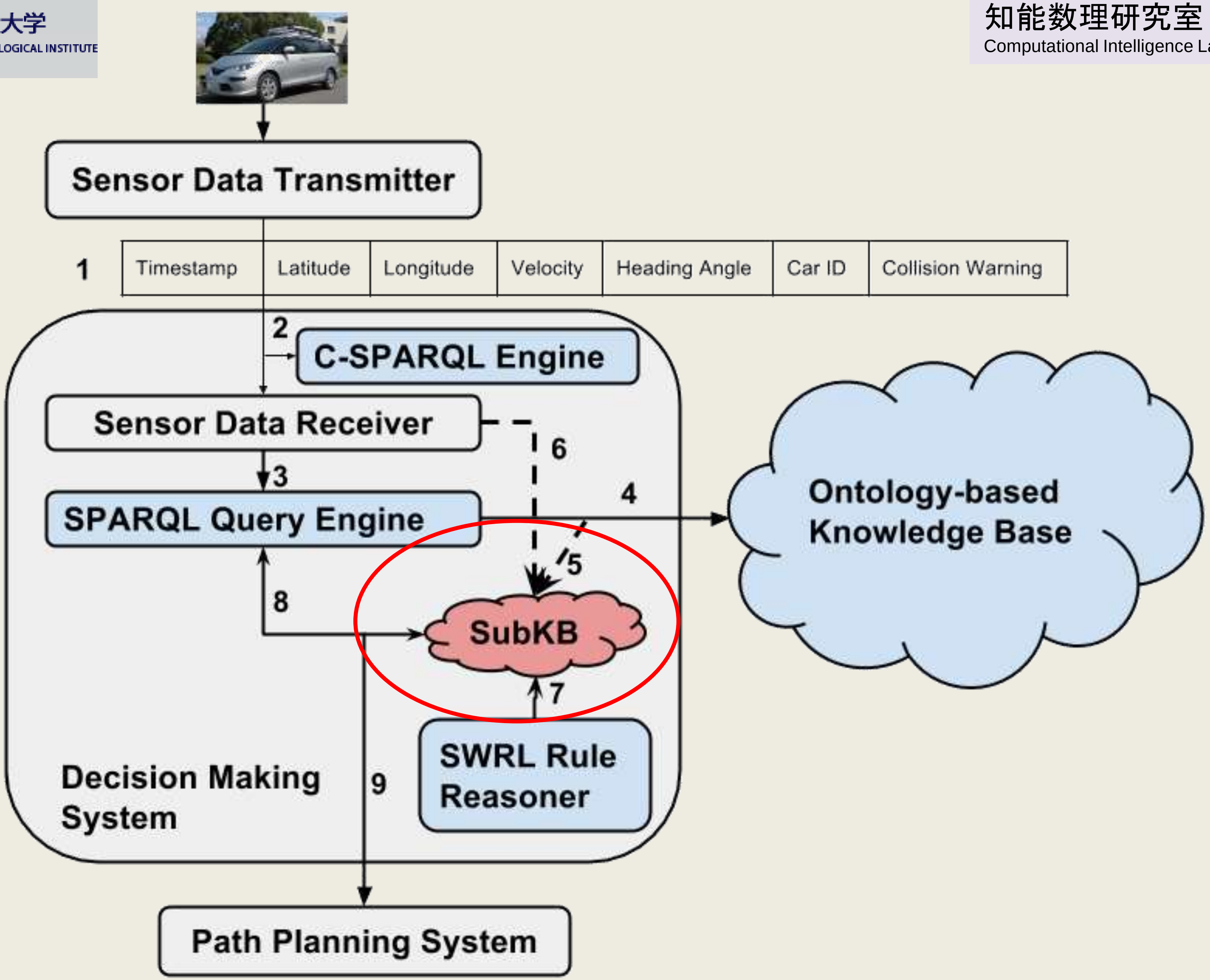


Case analysis



SWRL Rules (out of 14 rules)

1	collisionWarningWith(?carX, ?carY) ⇒ CollisionWarning(?carX) ∧ CollisionWarning(?carY)
2	Intersection(?int) ∧ isRunningOn(?carX, ?lane1) ∧ turnRightTo(?lane1, ?lane2) ∧ nextPathSegment(?lane1, ?int) ∧ nextPathSegment(?int, ?lane2) ⇒ TurnRight(?carX)
3	CollisionWarning(?carX) ∧ CollisionWarning(?carY) ∧ GoForward(?carY) ∧ TurnRight(?carX) ⇒ Stop(?carX) ∧ giveWay(?carX, ?carY)
4	MyCar(?car1) ∧ isRunningOn(?car1, ?int) ∧ Intersection(?int) ∧ collisionWarningWith(?car1, ?car2) ⇒ Stop(?car1) ∧ giveWay(?car1, ?car2)
5	TwoWayLane(?lane) ∧ isRunningOn(?carX, ?lane) CollisionWarning(?carX) ∧ CollisionWarning(?carY) ⇒ ToLeft(?carX), giveWay(?carX, ?carY)


Sensor Data Transmitter
1
Timestamp
Latitude
Longitude
Velocity
Heading Angle
Car ID
Collision Warning
2
C-SPARQL Engine
Sensor Data Receiver
6
3
SPARQL Query Engine
4
5
Ontology-based Knowledge Base
8
SubKB
7
Decision Making System
9
SWRL Rule Reasoner
Path Planning System

```
Input   : currRS # Current Road Segment
Output: SubKB # Sub-Knowledge Base
dirList ← getConnectedRS(currRS);
rsList ← dirList;
SubKB ← ∅
foreach rs ∈ dirList do
 |   rsList.add( getConnectedRS(rs) );
end
foreach rs ∈ rsList do
 |   SubKB.add( getAllInfo(rs) )
 |   if <rs, map:hasLane, lane> then
 |    |   laneList.add( lane )
 |   else
 |
 |   end
end
foreach lane ∈ laneList do
 |   SubKB.add( getAllInfo(lane) )
end
SubKB.add( SWRLRules )
return SubKB
```

Algorithm 1: Sub-Knowledge Base construction.

Sub-Ontology (SubKB)

Subject	Property	Object
yagoto:AnoNagoyaLine	rdf:type	map:PrefecturalRoad
yagoto:AnoNagoyaLine	map:hasIntersection	yagoto:AnoNagoyaInt3_4
yagoto:AnoNagoyaLine	map:hasRoadSegment	yagoto:AnoNagoyaRS3
yagoto:AnoNagoyaLine	map:hasRoadSegment	yagoto:AnoNagoyaRS4
yagoto:AnoNagoyaLine	map:speedMax	“40”^^kmh
yagoto:AnoNagoyaLine	map:osm_way_id	osm_way:122098916
yagoto:AnoNagoyaInt3_4	rdf:type	map:Intersection
yagoto:AnoNagoyaInt3_4	map:isConnectedTo	yagoto:AnoNagoyaRS3
yagoto:AnoNagoyaInt3_4	map:isConnectedTo	yagoto:AnoNagoyaRS4
yagoto:AnoNagoyaInt3_4	map:isConnectedTo	yagoto:GrandirLaneAdapter1
yagoto:AnoNagoyaInt3_4	map:boundPos	35.134697, 136.964103
yagoto:AnoNagoyaInt3_4	map:boundPos	35.134762, 136.964181
yagoto:AnoNagoyaInt3_4	map:boundPos	35.134788, 136.964072
yagoto:AnoNagoyaRS4	rdf:type	map:RoadSegment
yagoto:AnoNagoyaRS4	map:isConnectedTo	yagoto:AnoNagoyaInt3_4
yagoto:AnoNagoyaRS4	map:isConnectedTo	yagoto:AnoNagoyaCrossWalk1
yagoto:AnoNagoyaRS4	map:boundPos	35.134697, 136.964103
yagoto:AnoNagoyaRS4	map:boundPos	35.134574, 136.964147
yagoto:AnoNagoyaRS4Lane2	rdf:type	map:OneWayLane
yagoto:AnoNagoyaRS4Lane2	map:isLaneOf	yagoto:AnoNagoyaRS4
yagoto:AnoNagoyaRS4Lane2	map:enterPos	35.134570, 136.964125
yagoto:AnoNagoyaRS4Lane2	map:exitPos	35.134693, 136.964082
yagoto:AnoNagoyaRS4Lane2	control:turnRightTo	yagoto:GrandirLaneAdapter1
yagoto:AnoNagoyaRS4Lane2	control:goStraightTo	yagoto:AnoNagoyaRS3Lane2

Real time performance

Acceleration by sub-ontology

- Whole Ontology (Map)
  - Max: 965ms
  - Min: 305ms
  - Avg: 470ms
- Sub Ontology (Sub-Map)
  - Max: 236ms
  - Min: 37ms
  - **Avg: 53ms**

Size of Knowledge Base

- **Whole Knowledge Base**
  - 407kb (or larger)
- **Sub-Knowledge Base**
  - 19 ~ 40kb

Criticism?

- There are always some criticism to our approach.
  - Old style
  - Knowledge-base is slow
  - Poor handling of exceptional cases
  - Following a human way would lead to unsafety
  - Deep learning is more powerful and practical

It is time for the semantic technology community to tackle the real-world problems related to self-driving.

Our on-going studies:
Evaluations of Driving Ontologies

Quality evaluation of driving ontologies

- What are the conditions by which authorities approve level 5 cars to run on any public roads?

  → Open question!

- Idea
  - If an autonomous car passes its driver's license test, we can say the autonomous car has ability and knowledge to drive.

Overview

Paper test

Q “The traffic sign includes the instruction sign and the regulation sign.” (Yes/No)

Logical form

$\lambda x.\lambda y.\lambda z.\mathrm{TrafficSign}(x) \wedge \mathrm{InstuctionSign}(y) \wedge \mathrm{RegulationSign}(z) \wedge \mathrm{subClassOf}(y, x) \wedge \mathrm{subClassOf}(z, x)$

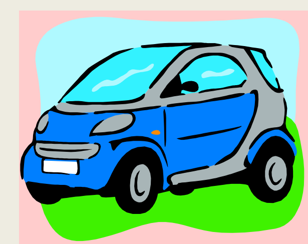
Driving test

SPARQL query

query

Ontology

query



Ontology-based driving

```
ASK WHERE {
  InstructionSign rdfs:subClassOf TrafficSign.
  RegulationSign rdfs:subClassOf TrafficSign .
}
```

Questions beyond the scope

- Commonsense question
  - "You can put boxes on a road" (Yes/No)
- Calculation question
  - "3 adults and 3 children can ride the car with the riding seats of 5." (Yes/No)
- Questions with figures

Our on-going studies:
Driving Ontology Expansion

Adding knowledge to TTI core ontology

Converting a driving guideline to knowledge.

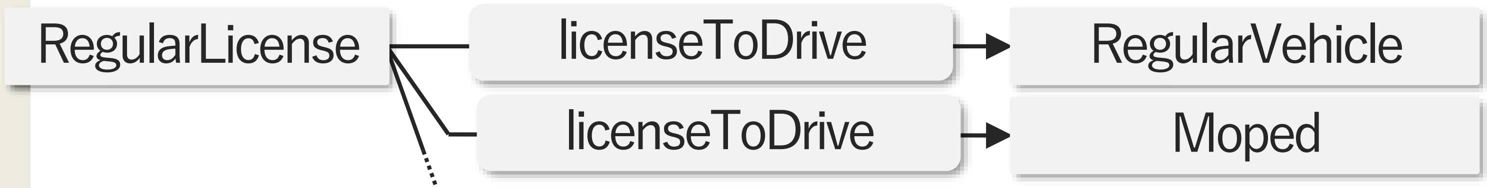
Guideline

You can drive the following types of vehicles ···

License type	car type
LargeLicense	···
MiddleLicense	···
RegularLicense	RegularVehicle, Moped, ···

……

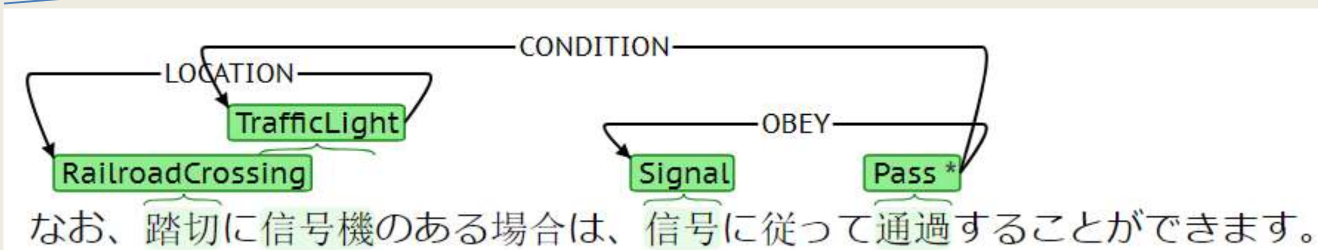
Ontology



Knowledge acquisition

- Manual semantic annotations of “Rules of Road”（交通教則）

  → on going



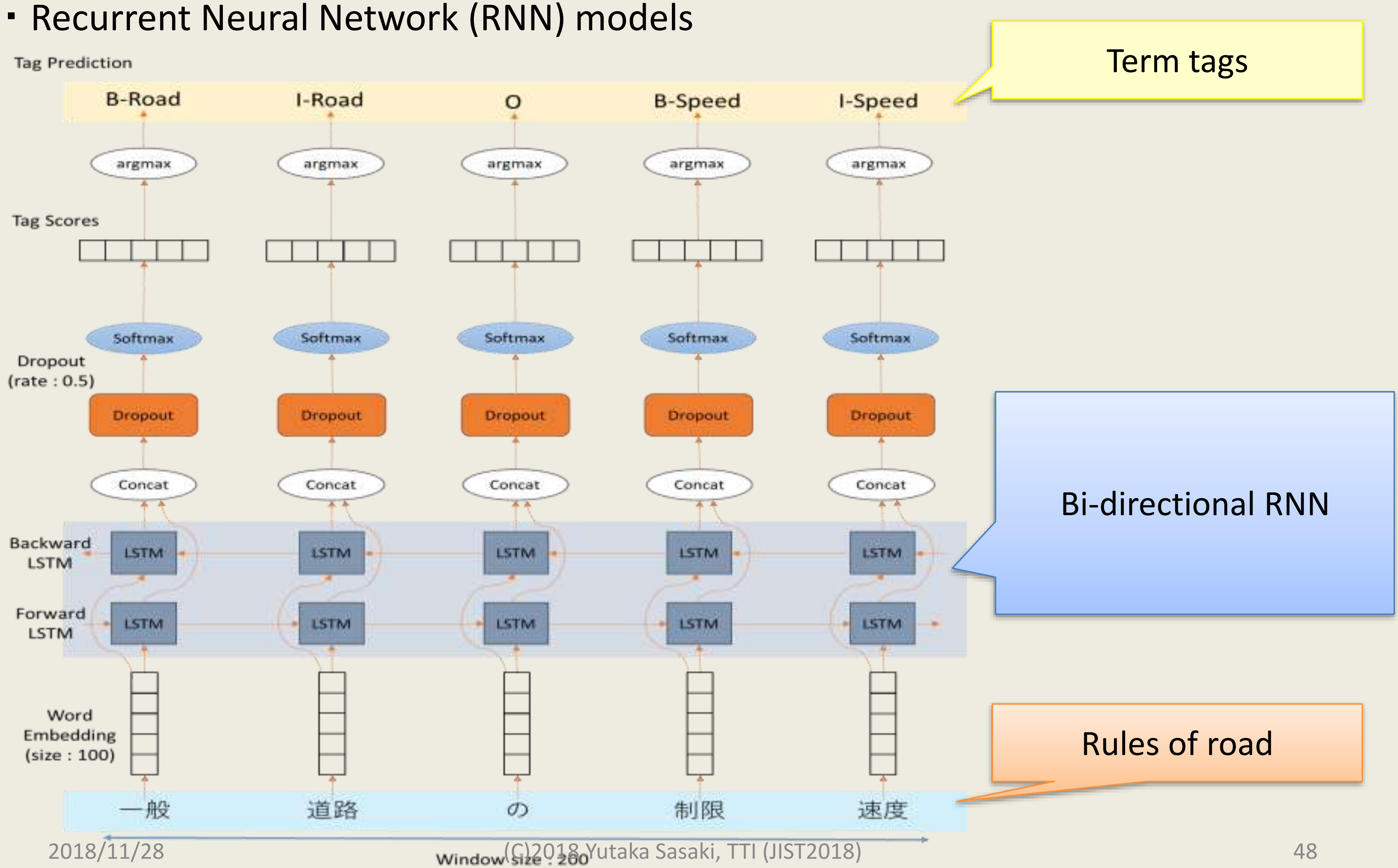
(If there is a traffic light at a railroad crossing, (you) can pass the crossing according to the signal.)

Deep Learning models

Semi-automatic ontology expansion

Term extraction

- Recurrent Neural Network (RNN) models



Experimental settings

Textual data:

Annotated “Rules of Road”

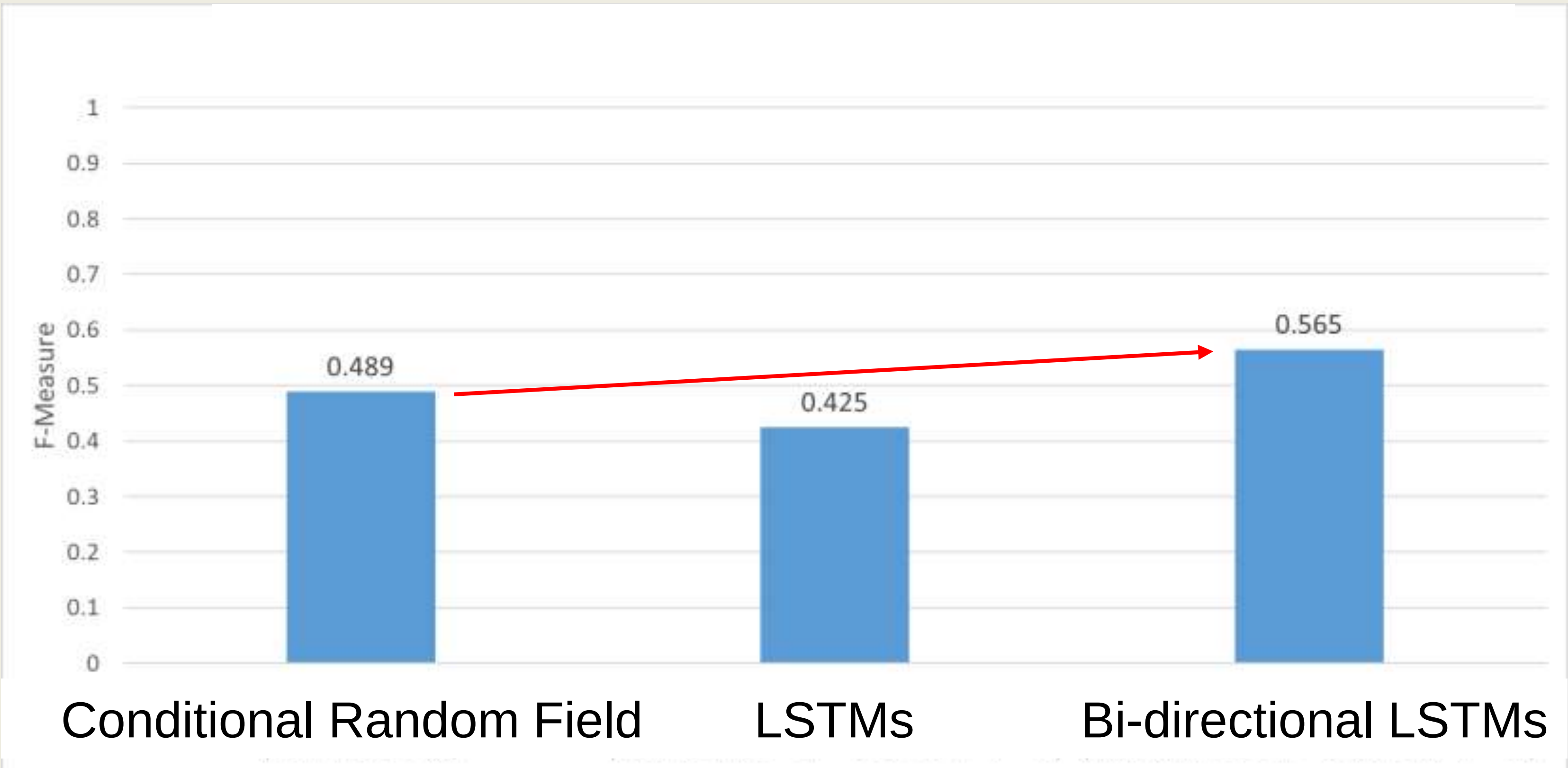
- #sentence : 2,150
- #words : 43,395
- #traffic terms: 8,822
- #traffic term hierarchy: 8 levels deep
- #distinct terms: : 780

Learning Model:

RNN

- dimensionality : 100
- batch size : 10
- iterations: 20,000
- type : LSTM (Long Short-Term Memory)

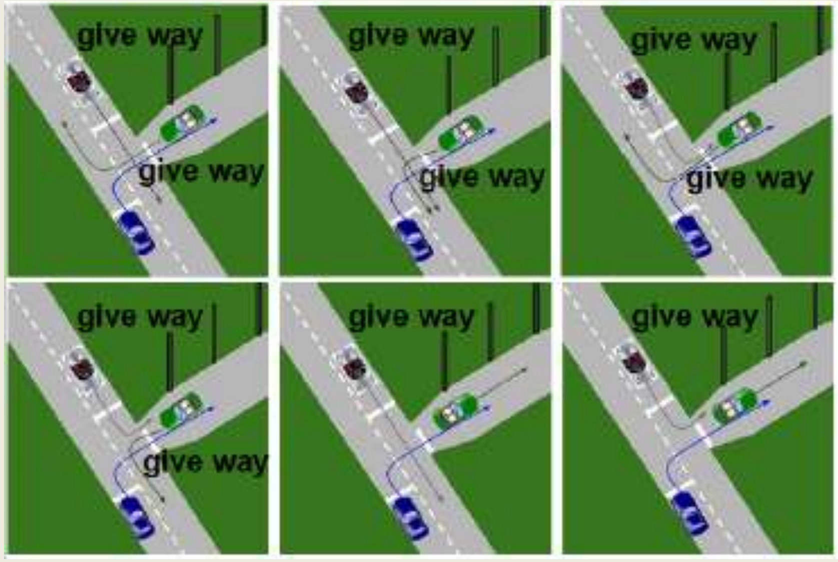
Term extraction results



$$\text{F-measure} = \frac{2*Precision*Recall}{Presicion+Recall}$$

Further uncontrolled cases

- Our previous approach targeted intersections.



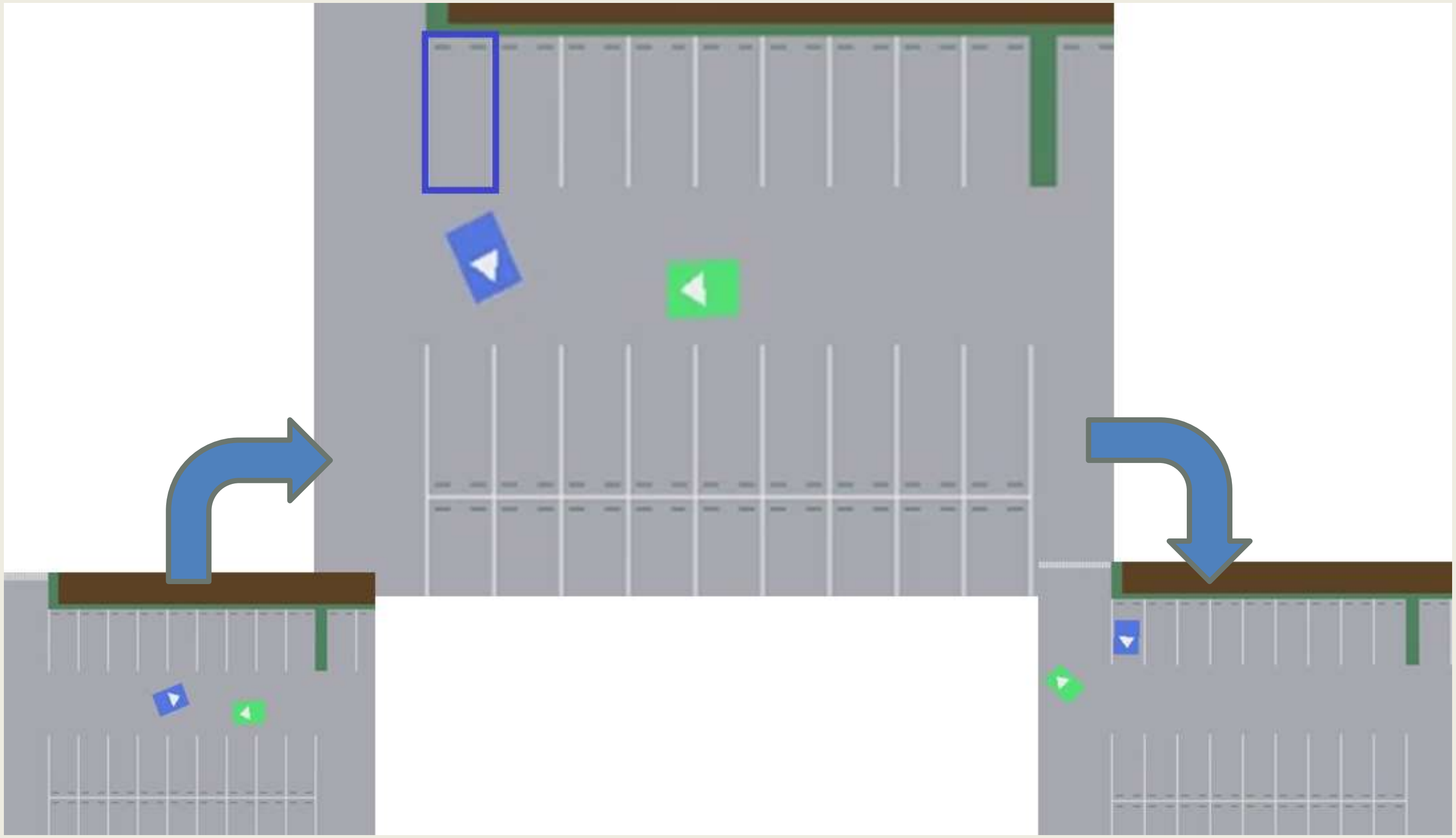
- In a parking lot, we need a different approach.
  → Less restricted

  - Our own simulator for parking scenario.
  - Ontology rules should include parking manner, not laws.

Current progress

- Parking scenario: manner-based decisions

IN/OUT



Conclusions

- The ontology-based driving approach leads to the level 5 self-driving.
- A lot of open questions. Studies are ongoing.
- Reasoning time is an issue:
  - To accelerate, the dynamic creation of a sub-ontology is effective.
- Semantic technology will be a core technology for the level 5 self-driving.

Acknowledgements

Collaborators

Lihua Zhao, Ryutaro Ichise, Zheng Liu, Sei-ichi Mita, Makoto Miwa, Tatsuya Yoshikawa, Takeshi Naito, Toshiaki Kakinani,

and the past and current postgraduate students at the COIN Lab. Naonori Nagano, Katana Suenaga, Junya Takayama, Zheming Lyu