Designing and Publishing Illegally Parked Bicycle LOD

Shusaku Egami * , Takahiro Kawamura † , Akihiko Ohsuga ‡

Abstract

The illegal parking of bicycles is an urban problem in Japan and other countries. Illegally parked bicycles (IPBs) obstruct vehicles and pedestrian traffic, cause road accidents, encourage thefts, and disfigure streets. To solve the challenge posed by IPBs, we believed that it is necessary to collect and republish the data as reusable format. Therefore, we collected the number of IPBs, location information, time, and factors. Then, we integrated and republished these data as Linked Open Data (LOD) on the Web. In this paper, we described a schema design of illegally parked bicycle LOD (IPBLOD) and a methodology of designing LOD schema. Then, we collected data from SNS and website of municipality, and built the LOD of 98,982 triples. Furthermore, we presented several statistics based on IPBLOD.

Keywords: Linked Open Data, Semantic Web, Ontology, Urban problem.

1   Introduction

The illegal parking of bicycles around railway stations is an urban problem in Japan. An increase in awareness of health problems [1] and energy conservation [2] led to a 2.6 fold increase in the number of bicycle ownership in Japan from 1970 to 2013. In addition to the insufficient availability of bicycle parking spaces, inadequate public knowledge on bicycle parking laws has contributed to this urban problem. The IPBs obstructs vehicles and pedestrian traffic, cause road accidents, encourage thefts, and disfigure streets.

Thus, to address this challenge, we realized that it is necessary to publish the distribution of IPBs as Open Data[1]. Also, it is recommended that the Open Data should be structured according to the Resource Description Framework (RDF) and that relevant links should be created between the data elements. This data set is called Linked Open Data (LOD). By using LOD on illegally parked bicycles (IPBs), for example, visualization of the IPBs, suggestion of locations for optimal bicycle parking spaces, removal of IPBs, and urban design will be possible. However, Open Data sets available for IPBs are currently distorted, and it is difficult to exploit. In addition, other data concerning issues such as bicycle parking spaces and government statistics, have been published in a variety of formats. Hence, a unification of data formats and definition of schema for data storage are important issues

---

* The University of Electro-Communications, Tokyo, Japan
† Japan Science and Technology Agency, Tokyo, Japan
‡ The University of Electro-Communications, Tokyo, Japan
[1]`http://opendatahandbook.org/guide/en/what-is-open-data/`

Table 1: Results of clustered keywords

Property	Category	Keyword
place and factor	POI	shopping street, large-scale retail stores, public office, school, library, public hall, department store, supermarket, bank, amusement, pachinko, commercial building
	bicycle parking	usability, safety, conforts, supply, fee
	train	train, distance from home to the station
	storage space	capacity
time and factor	time	weekday, holiday, hour
product	accident	congestion, blocking vegicle traffic
factor	objective	commutation, destination
	weather	rainy
	bicycle	price reduction, deterioration, cost of maintenance, variety

that need to be addressed. In this paper, we described a schema design of illegally parked bicycle LOD (IPBLOD). Then, we described an approach of building LOD based on the data collected from SNS and the websites of municipalities.

The remaining sections of this paper is organized as follows. In Section 2, the methodology of designing LOD schema and IPBLOD are presented. In Section 3, the statistics and utility of the schema are discussed. In Section 4, schema design and data collection method are described. Finally, Section 5 concludes this paper with some feasible future extensions.

2   Building IPBLOD

In this study, we design the practical LOD schema based on the existing ontologies. In this section, we explain our methodology of designing LOD schema for day-to-day urban problems. Then, the data collection method and building LOD are described.

2.1   A methodology for designing LOD schema

Illegally parked bicycles are observed by social sensor, since it is difficult to install sensors in the streets. In our previous work [10], the schema of IPBLOD was based on the SSN. Figure 1 shows IPBLOD schema based on SSN. However, in order to address this urban problem using LOD, the LOD should not only have the number of IPBs, location and time information, but also contain the factors of IPBs, such as POIs and weather. In this paper, we present the LOD schema, including factors of IPBs, and we propose the methodology for designing LOD schema on urban problems, such as IPBs.

In the ontology study, the methodology for building ontology has been discussed. We propose the methodology for designing practical LOD schema in reference to Activity-First Method [12]. The schema of IPBLOD is designed based on this methodology, which consists of two steps as follows:
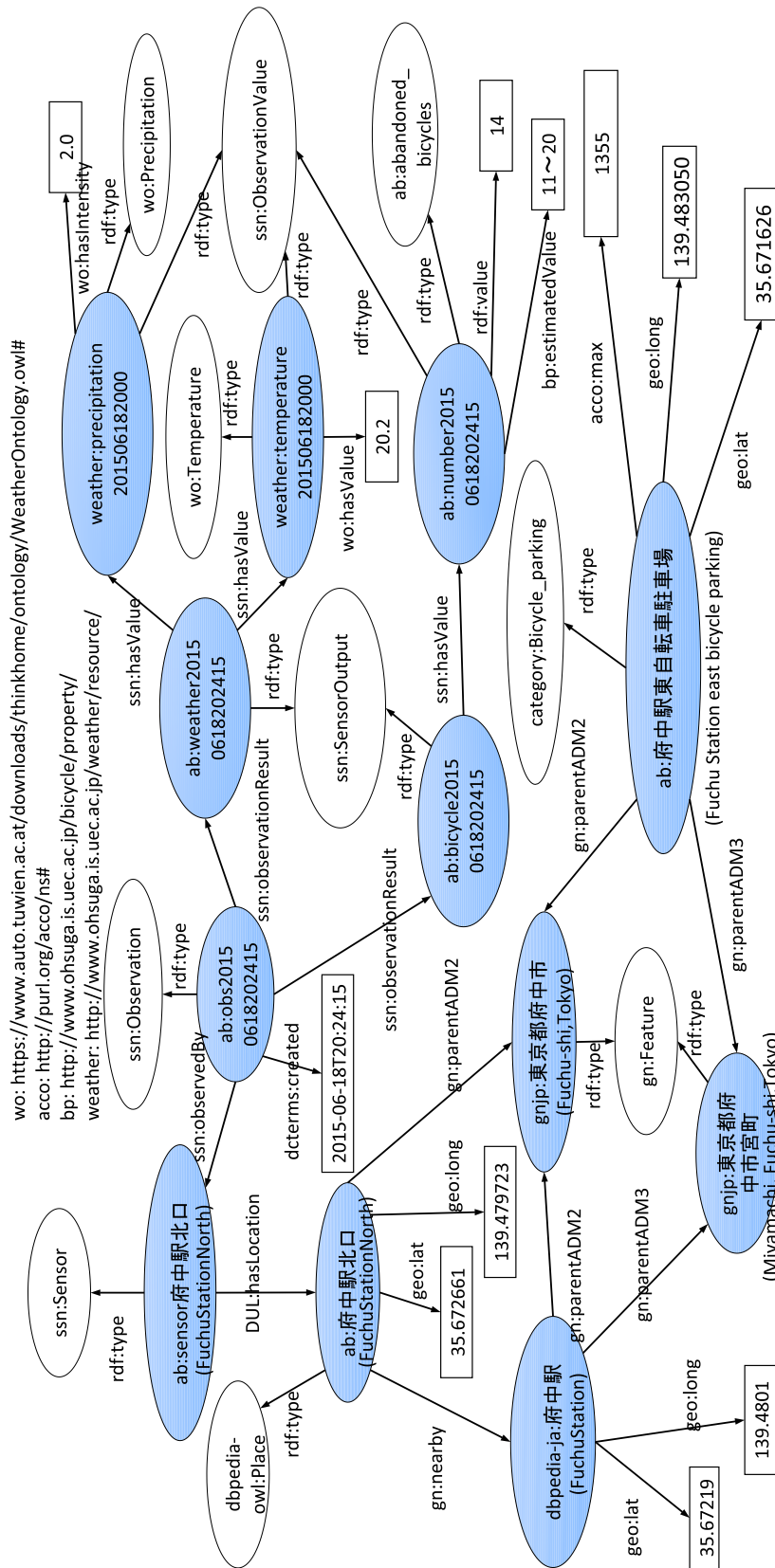
Figure 1: LOD schema based on SSN

1. Extraction of domain requirements

   a. Select an ontology that models the urban problem

   b. Search for articles on the urban problem using a search engine

   c. Extract keywords from the articles based on properties of the ontology

   d. Cluster the keywords

2. Designing schema

   a. Design classes based on the ontology

   b. Design instances and properties based on the result of the clustering

First, the existing ontology is selected in order to build LOD based on an ontology. To solve the challenge posed by IPBs using LOD, it is necessary to consider the practicality as well as the semantic consistency. Therefore, we selected Event Ontology (EO)[2], as a practical structure, where IPBs can be considered as an event. In the EO, an event class has properties on place, time, agent, factor, and product.

Next, we searched for articles on IPBs using Google. Then, we investigated the top 10 articles and their references, and then extracted the keywords based on the properties of the existing ontology. Specifically, keywords are extracted from sentences that describe the place, time, active, agent, factor, and product.

The extracted keywords are clustered manually as in Table 1. Then, the classes are designed based on the EO. The expression of Description Logic (DL) is as follows:

$$IllegallyParkedBicycles \sqsubseteq Event$$

$$IllegallyParkedBicycles \sqsubseteq \exists place.SpatialThing$$

$$IllegallyParkedBicycles \sqsubseteq \exists time.TemporalEntity$$

$$IllegallyParkedBicycles \sqsubseteq \exists weather.WeatherState$$

$$IllegallyParkedBicycles \sqsubseteq \exists factor.Thing$$

$$IllegallyParkedBicycles \sqsubseteq \exists agent.Agent$$

$$IllegallyParkedBicycles \sqsubseteq \exists product.Thing$$

$$IllegallyParkedBicycles \sqsubseteq \exists value.Integer$$

The IllegallyParkedBicycles class refers to the aggregate of IPBs, and it is a subclass of the Event class. The IllegallyParkedBicycles has the place, time, weather, agent, factor, product, and the number of IPBs.

Then, we designed instances in reference to Table 1. Figure 2 shows an overview of LOD schema with the instances. In Table 1, the column of category refers to the instance, and the instance is linked with the EO property. Also, some instances are linked to other instances. However, it is difficult to get the data on people, who parked their vehicle illegally. Therefore, we omitted the EO agent property. Also, we ommited the product classes, since it is difficult to obtain the data on accidents caused by IPBs. In the same way, the storage space, the objective of the person who parked the bicycle illegally, and the price of the bicycle are also omitted. Moreover, the Point Of Interest (POI), the nearest bicycle parking,
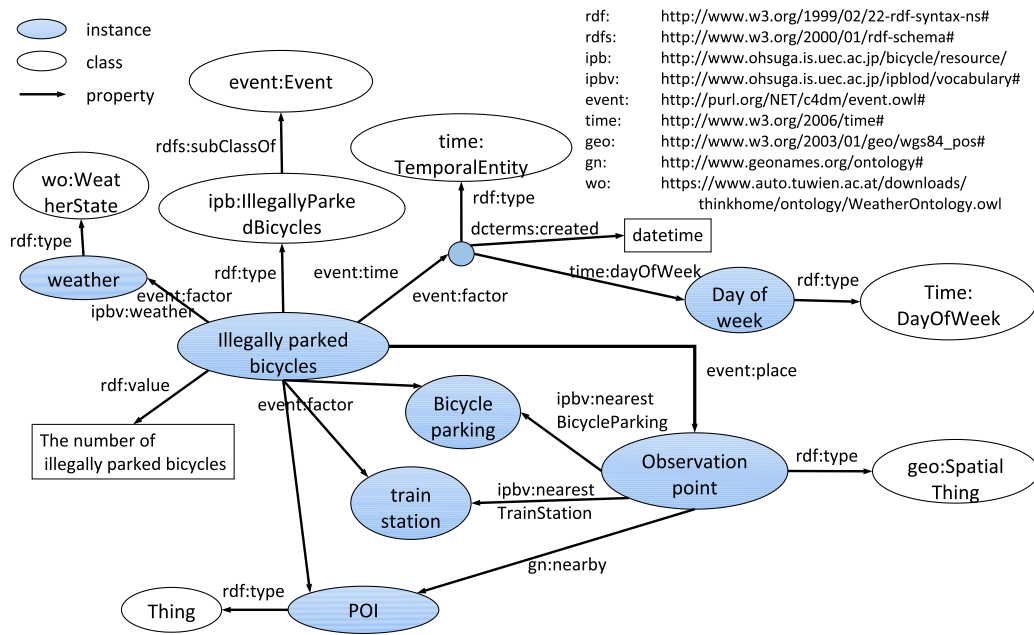
---

[2]`http://purl.org/NET/c4dm/event.owl`

Figure 2: LOD schema containing instances

the nearest train station, time, and the weather are added to the LOD schema as factors of IPBs. The nearest train station is a resource of DBpedia Japanese[3]. A place, which is an observation point, and a POI schema are designed as follows:

Place $\sqsubseteq \exists$lat.Decimal

Place $\sqsubseteq \exists$long.Decimal

Place $\sqsubseteq \exists$geometry.Geometry

Place $\sqsubseteq \exists$prefecture.Feature

Place $\sqsubseteq \exists$city.Feature

Place $\sqsubseteq \exists$town.Feature

Place $\sqsubseteq \exists$street.Feature

Place $\sqsubseteq \exists$nearby.POI

Place $\sqsubseteq \exists$nearestTrainStation.TrainStation

Place $\sqsubseteq \exists$nearestBicycleParking.BicycleParking

Geometry $\sqsubseteq \exists$serialization.Literal

POI $\sqsubseteq \exists$lat.Decimal

POI $\sqsubseteq \exists$long.Decimal

POI $\sqsubseteq \exists$geometry.Geometry

The Place and POI have a latitude, a longitude, and a geometry instance. Since a geometry instance has location information as Well-known text (WKT) serialization format,

---

[3]http://ja.dbpedia.org

it is possible to query using GeoSPARQL[4]. Also, the Place has a prefecture instance, a city instance, a town instance, and a street instance. Since the Place has these instances, it is possible to query statistics by area and link other statistics such as population and population density. In order to describe these relations, we use properties of GeoNames Ontology[5]. In terms of the relations of the nearest bicycle parking space and the nearest train station, we define new properties. A bicycle-parking schema is designed as follows:

$$BicycleParking \sqsubseteq \exists lat.Decimal$$

$$BicycleParking \sqsubseteq \exists long.Decimal$$

$$BicycleParking \sqsubseteq \exists geometry.Geometry$$

$$BicycleParking \sqsubseteq \exists prefecture.Feature$$

$$BicycleParking \sqsubseteq \exists city.Feature$$

$$BicycleParking \sqsubseteq \exists town.Feature$$

$$BicycleParking \sqsubseteq \exists street.Feature$$

$$BicycleParking \sqsubseteq \exists monthlyFee.UnitPriceSpecification$$

$$BicycleParking \sqsubseteq \exists dailyFee.UnitPriceSpecification$$

$$BicycleParking \sqsubseteq \exists max.Integer$$

$$Geometry \sqsubseteq \exists serialization.Literal$$

The BicycleParking has also geographical information. Furthermore, the BicycleParking has parking fees and capacity. The time schema is designed based on the Time Ontology[6] as follows:

$$time : TemporalEntity \sqcap time : DateTimeDescription$$

$$TemporalEntity \sqsubseteq \exists created.DateTime$$

$$DateTimeDescription \sqsubseteq \exists year.Integer$$

$$DateTimeDescription \sqsubseteq \exists month.Integer$$

$$DateTimeDescription \sqsubseteq \exists day.Integer$$

$$DateTimeDescription \sqsubseteq \exists hour.Integer$$

$$DateTimeDescription \sqsubseteq \exists minute.Integer$$

$$DateTimeDescription \sqsubseteq \exists second.Integer$$

$$DateTimeDescription \sqsubseteq \exists dayOfWeek.DayOfWeek$$

The range of event:time property must be an instance of TemporalEntity class. In the Time Ontology, domains of time:year, time:month, time:day, time:hour, time:minute, time:second, and time:dayOfWeek properties must be an instance of the DateTimeDescription class. Therefore, we defined that time is an instance of the TemporalEntity and the DateTimeDescription. The weather schema is designed based on the Weather Ontology[7] as follows:
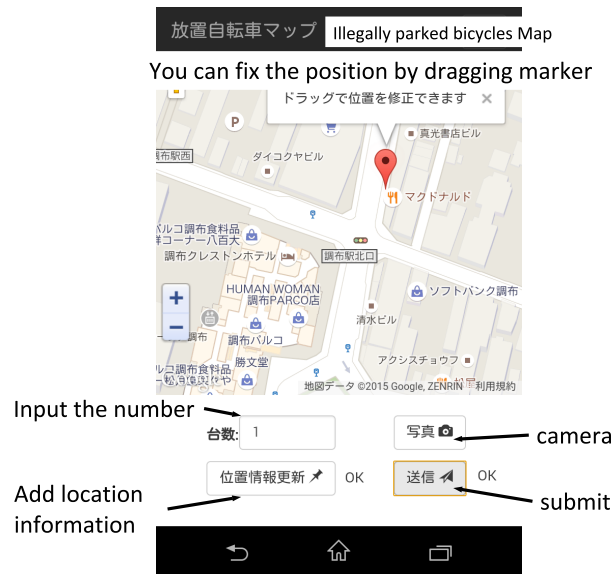
---

[4] https://www.w3.org/2011/02/GeoSPARQL.pdf

[5] http://www.geonames.org/ontology/documentation.html

[6] https://www.w3.org/TR/owl-time

[7] https://www.auto.tuwien.ac.at/downloads/thinkhome/ontology/WeatherOntology.owl

Figure 3: Screenshot of the tweet application

WeatherState ⊑ ∃ hasWeatherPhenomenon.Precipitation

WeatherState ⊑ ∃ hasWeatherPhenomenon.Temperature

Precipitation ⊑ ∃ hasIntensity.Float

Temperature ⊑ ∃ hasValue.Float

WeatherState ⊑ ∃ hasObservationTime.PropertInterval

ProperInterval ⊑ ∃ hasBeginning.InstantOfTime

ProperInterval ⊑ ∃ hasEnd.InstantOfTime

Instant ⊑ ∃ inXSDDateTime.DateTime

WeatherState ⊑ ∃ observedBy.MeteorologicalSensor

MeteorologicalSensor ⊑ ∃ hasLocation.Feature

The WeatherState has weather phenomena. In this schema, a weather phenomenon is precipitation or temperature. The WeatherState is observerd by a meteorological sensor.

Thus, after designing the classes of LOD based on the existing ontology, the extracted keywords are added to the LOD as instances. Furthermore, if the LOD that contains the factors of IPBs does not exist, the schema of the factor is designed.

2.2   Collection of observation data

We started this study by with collection of tweets containing location information, pictures, hash-tags, and the number of IPBs. However, obtaining the correct locations from Twitter is difficult, since mobile phones often attach incorrect location information. Mobile phones are equipped with inexpensive GPS chips, and it is known that the accuracy will be inaccurate due to weather conditions and GPS interference area [13]. To address this problem, we developed a web application that enables users to post tweets on Twitter after correcting their location information, and we made an announcement asking public users to post
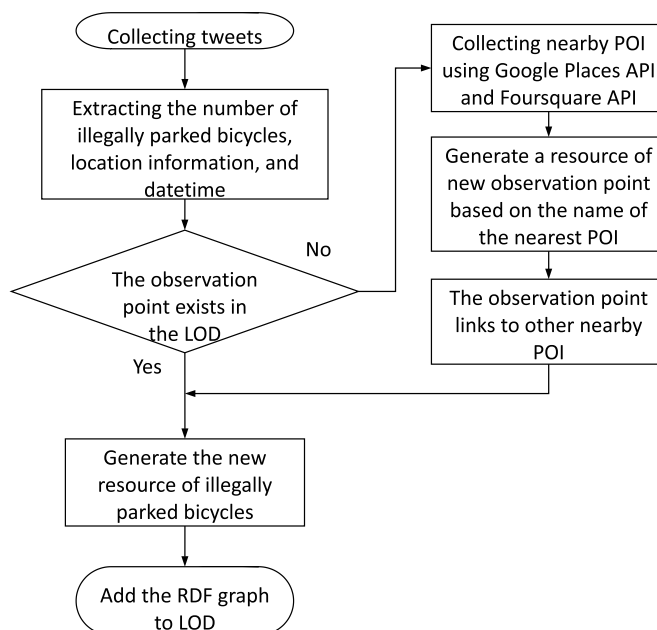
Figure 4: Process of LOD building

tweets of IPBs using this application. Figure 2 shows a screen shot of this application. After OAuth authentication, a form and buttons are shown. When the location button is pressed, a marker is displayed at the user's current location on a map. The marker is draggable, thus allowing users to correct their location information. When the users add their location information, enter the number of IPBs, take pictures, and submit them, then tweets including this information with a hashtag are posted.

Furthermore, we collected information on POI using Google Places API[8] and Foursquare API[9]. Also, we obtained bicycle parking information from websites of municipalities, and retrieved weather information from the website of the Japanese Meteorological Agency (JMA)[10].

2.3   Building LOD based on designed schema

The collected data on IPBs are converted to LOD based on the designed schema. Figure 4 shows the process of building IPBs. First, the server program collects tweets containing the particular hash-tags, the location information, and the number of IPBs in real-time. The number of IPBs is extracted from the text of tweets using regular expressions.

Next, the server program checks whether there is an existing observation point to a radius of less than 30m by querying SPARQL endpoint[11] using the SPARQL query as follows.

---

[8]https://developers.google.com/places/?hl=ja
[9]https://developer.foursquare.com/
[10]http://www.jma.go.jp/jma/indexe.html
[11]http://www.ohsuga.is.uec.ac.jp/sparql
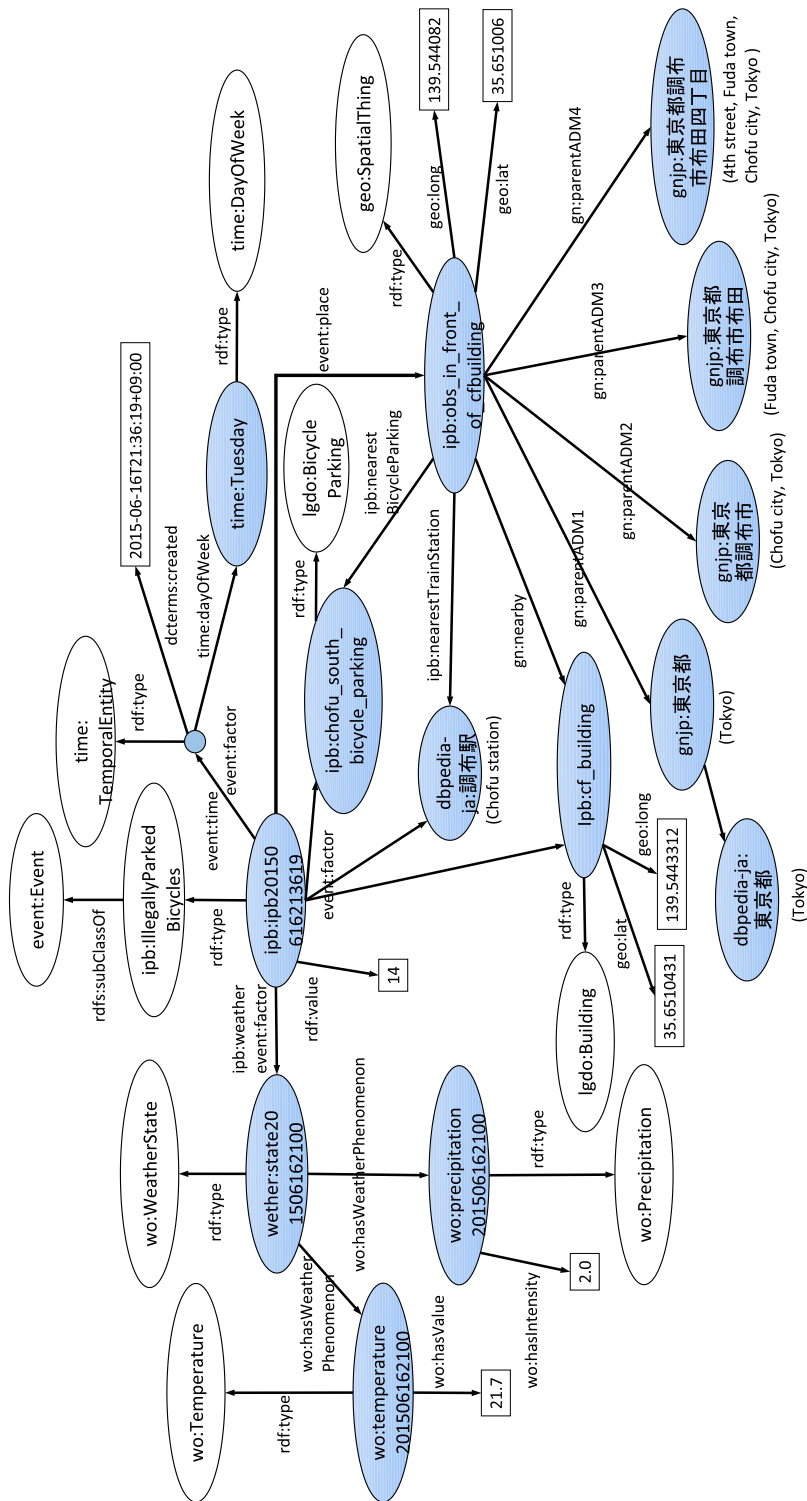
Figure 5: Part of the integrated LOD

```
PREFIX rdf: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX geo: <http://www.w3.org/2003/01/geo/wgs84_pos#>
PREFIX ngeo: <http://geovocab.org/geometry#>
PREFIX ogcgs: <http://www.opengis.net/ont/geosparql#>
SELECT ?s ?lat ?lng
FROM <http://www.ohsuga.is.uec.ac.jp/ipblod>
WHERE {
        ?s rdf:type geo:SpatialThing .
        ?s ngeo:geometry [
                ogcgs:asWKT ?g
        ];
                geo:lat ?lat;
                geo:long ?lng;
        FILTER(
                <bif:st_intersects> (
                ?g, <bif:st_point> (lat, lng), radius
                )
        )
}
```

If there is no observation point on the IPBLOD, the point is added as a new observation point. In order to add new observation points, the nearest POI information is obtained using Google Places API and Foursquare API. The new observation point is generated based on the name of the nearest POI. It is possible to obtain the types of the POI from Google Places API and Foursquare API. We mapped the types of POI to classes in LinkedGeoData[12]. Thus, the POI is an instance of classes in LinkedGeoData. However, some POI have not been attached properly types. Therefore, their types are decided by keyword search with the name of the POI. There are same POIs in Google Places API and Foursquare API. However, there are cases that names of those POIs are not same. Thus we integrated those POIs using a similarity calculated by Eq.1 based on the edit distance. n_a and n_b are names of POIs. $EditDist(n_a, n_b)$ is the edit distance of n_a and n_b. The edit distance is the minimum number of editing operations needed to make n_a and n_b the same string. $Max(n_a, n_b)$ is the larger length of two strings between the n_a and n_b. If the similarities became greater than 0.8, we integrated two POIs.

$$sim(n_a, n_b) = 1 - \frac{EditDist(n_a, n_b)}{Max(n_a, n_b)} \qquad (1)$$

Then, the address, prefecture's name, and city name are obtained using Yahoo! reverse geocoder API, and then the links to GeoNames.jp[13] are generated based on the obtained information. GeoNames.jp is a Japanese geographical database. Thus, data are collected and added to the IPBLOD using Web APIs in real time. Figure 5 shows part of the IPBLOD. The LOD is stored in Virtuoso Open-Source Edition [14], which is an open source RDF database, and is published via a SPARQL endpoint[15]. Also, the RDF dataset is published with CC-BY license on our website[16].

Table 2: Statistics for observation data

Area	#observation points	Amount of observation data
Chofu-shi, Tokyo	29	716
Nerima-ku, Tokyo	4	178
Fussa-shi, Tokyo	2	84
Chuo-ku, Sapporo-shi, Hokkaido	46	56
Naka-ku, Yokohama-shi, Kanagawa	1	39
Fuchu-shi, Tokyo	17	32
Shinjuku-ku, Tokyo	25	26
Musashino-shi, Tokyo	10	21
Shibuya-ku, Tokyo	5	5
Kokubunji-shi, Tokyo	3	4
Isogo-ku, Yokohama-shi, Kanagawa	2	3
Kita-ku, Sapporo-shi, Hokkaido	3	3
chuo-ku, Kobe-shi, Hyogo	3	3
Tama-shi, Tokyo	2	2
Naka-ku, Nagoya-shi, Aichi	2	2
Shinagawa-ku, Tokyo	1	1

3 Discussion

There are 1172 pieces of observational data collected in total from January 2015 to October 2016. The number of triples (records in DB) included in the IPBLOD was 98,982. Table 2 shows the statistics on the observation data. As a result, the observation data on Chofu city has the largest amount. Since we posted promotion tweets on our twitter accounts, contributors in our university increased.

In this study, we designed a practical LOD schema. Specificaly, we avoided hierarchical redundancy. Literals and numerical values can be obtained from the instance of IPBs class in three hops or less. Therefore, it is possible to get search results easily using short SPARQL query. For instance, we obtained the average number of IPBs per hour using the SPARQL query as follows.

```
SELECT ?hour (AVG(?value) AS ?average)
WHERE {
  ?s rdf:value ?value .
  ?s event:time/time:hour ?hour .
} GROUP BY ?hour
ORDER BY ASC(?hour)
```

Figure 6 shows a visualization of the result. We found from the result that there are

---

[12] http://linkedgeodata.org
[13] http://geonames.jp/
[14] http://virtuoso.openlinksw.com/
[15] http://www.ohsuga.is.uec.ac.jp/sparql
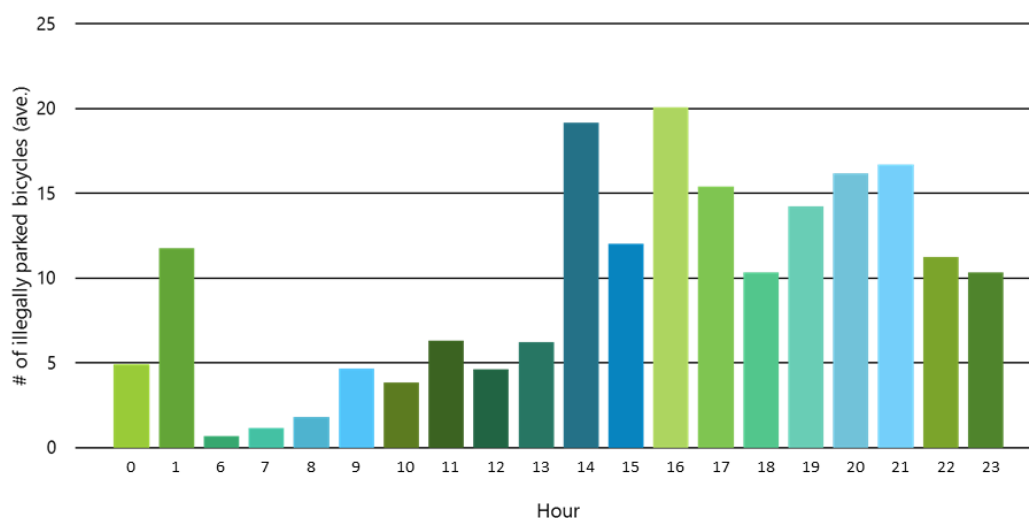[16] http://www.ohsuga.is.uec.ac.jp/bicycle/dataset.html

Figure 6: The average number of IPBs per hour

more IPBs in the night rather than the morning. In general, many bicycles are illegally parked during morning-commuting hours. However, the opposite result was shown in this study. Such results can be obtained, since the LOD schema is designed as detail time-series data.

Also, we obtained the statistics of POI types in the IPBLOD using the SPARQL query as follows.

```
SELECT DISTINCT ?ntype count(?ntype) WHERE {
  ?place a geo:SpatialThing .
  ?place gn:nearby ?near .
  ?near a ?ntype ;
} GROUP BY ?ntype
ORDER BY DESC(COUNT(?ntype))
```

Table 3 shows the top 10 POI types in the IPBLOD. As a result, we found that IPBs tend to exist around of restaurants, retail stores, and food shops.

Table 4 shows the statistics of the number of IPBs. In this study, since we did not define the range of observation points, there were differences in the range decisions for each person. It was found that some people tweeted many IPBs at one time, while some people divided the IPBs and tweeted individually. Therefore, the amount of 0-9 data tend to be the largest. To solve this problem, we will visualize a specified range of circles that indicate observation points in the tweet application.

Also, the statistic in Table 2 can be obtained easily using SPARQL, since the LOD schema is designed as geographically fine-grained data. Therefore, it is considered that the schema is designed useful for practical use. Furthermore, definition of various factors, such as time, weather, POI, etc. based on the Event Ontology could help address the actual causes of IPBs that cannot be found in a simple graph or table. However, the schema has some ontological inconsistency. Also, the detail level of each event:factor property values is

Table 3: Statistics for POI types

POI types	Amount of POI types
Restaurant	473
Retail	380
Food	366
Shop	365
Building	310
Health	116
Hairdresser	80
Hospital	63
Financial	59
Bar	58

Table 4: Statistics for the number of IPBs

Classification	# of IPBs
0-9	710
10-19	228
20-29	36
30-39	62
40-49	22
50-	16

different. There is a risk that reduces the utility of the LOD, since it follows the restrictions of ontologies and makes hierarchy redundant. In the future, we should consider designing new ontology, in order to design practical and ontologically-consistent schema. Also, we should consider the extension of the methodology in order to design LOD schema.

While, there are some noise data where bicycle might be not illegally parked in IP-BLOD. The noise data is caused by faulty operations. To avoid the noise data, we will use the data of no-parking zones. The no-parking zones are defined by municipalities, and if people parked bicycles in the zones, they must pay a fine. We considered that noise data can be excluded using the data of no-parking zones, and built the RDF data as a prototype based on the picture of no-parking zones around of Isogo Station. Figure 7 shows the schema of the no-parking zones. The schema is based on NeoGeo Geometry Ontology[17] and defined as polygon data. We stored this data in Virtuoso and visualized on a map (Figure 8) after obtaining the results using SPARQL. The red frame is a no-parking area, and red markers are observation points. In this way, it is possible to intuitively understand the relations between observation points and no-parking zones, and the exclusion of noise data can be implemented.

The IPBLOD fulfills the criteria of Linking Open Data cloud diagram[18]. Since the IPBLOD links to many other LOD indirectly, LOD has high utility. In the future, by inter-linking other LOD on urban problems with the IPBLOD, new values would be envisaged.

4   Related Work

Lawrynowicz et al. [3] proposed the hazardous situation ontology design pattern. This study presented a pattern using DOLCE+Dns [4], Event and Situation Ontology [5], and PSL ontology[19]. Deng et al. [6] proposed an ontology for citizens as sensors, and built Linked Data on road kill. This study used Friend Of A Friend (FOAF), Semantically-Interlinked Online Communities (SIOC) core ontology, and Semantic Sensor Network Ontology (SSN)[15]. The data are collected from Facebook using crowdsourcing. SOWL [16]

---

[17]http://geovocab.org/geometry.html
[18]http://lod-cloud.net
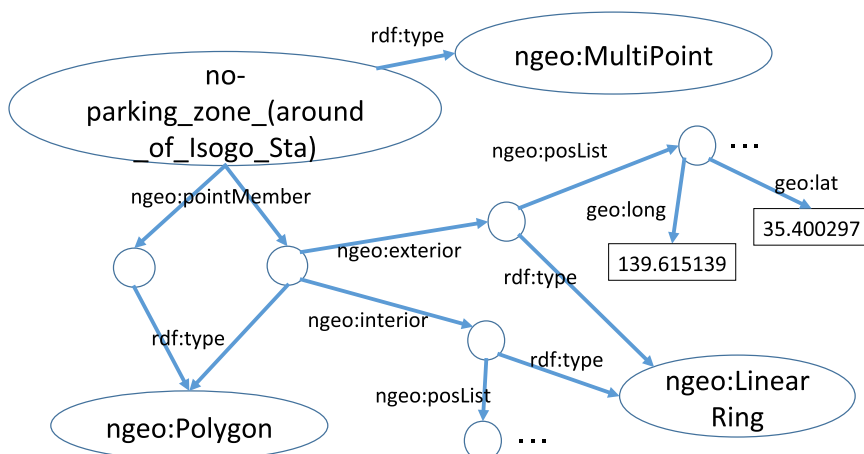[19]http://www.mel.nist.gov/psl/psl-ontology/

Figure 7: Schema of the no-parking zones around of Isogo Sta.

is a spatio-temporal data model for Web Ontology Language. In SOWL, temporal information is modeled based on the 4D-fluents approach [17] and spatial information is modeled based on region connection calculus and cone-shaped directional relations. Also, L3 [18] is proposed as a spatio-temporal data model for next-generation GISs. The L3 can track the evolution of spatial entities throughout time, and it is useful in the area of land use/land cover change (LULCC). However, in the urban problem area, the data model should be easy to understand in order to facilitate the data use in Civic Tech events. Therefore, we must build a new data schema for urban problems.

Data collection methods for building Open Data and Linked Data include crowdsourcing and Game With A Purpose (GWAP) [7]. A number of projects have employed these techniques. OpenStreetMap[20] is a project that creates an open map using crowdsourced data. Users can edit the map, and the data are published as Open Data. FixMyStreet[21] is a platform for reporting regional problems, such as road conditions and illegal dumping. Crowdsourcing, a method used for collecting information on FixMyStreet, has shown that regional problems can be solved more quickly than ever before. Zook et al. [8] reported a case, in which the crowdsourcing was used to link published satellite images with OpenStreetMap after the Haitian Earthquake. A map for the relief effort was created, and the data were published as Open Data. Celino et al. [9] have proposed an approach for editing and adding Linked Data using a Game with a Purpose (GWAP) and Human Computation. However, since the data concerning IPBs are time-series data, it is difficult to collect data using these approaches. Therefore, we collected data from SNS and the websites of municipalities.

Since our LOD was temporally sparse data, we estimated and complemented temporal missing data using Bayesian Networks [11]. In addition, we developed a web application visualizing time-series changes of IPBs in order to raise the awareness of the issue in local residents.

---

[20]`https://www.openstreetmap.org`
[21]`http://fixmystreet.org/`

Figure 8: Result of visualization of the no-parking zones around of Isogo Sta.

5 Conclusion

In this paper, we described schema design of the IPBLOD. The methods of collecting and converting data to LOD based on the designed schema was also presented. We can build other practical LOD on urban problems using the proposed methodology for designing LOD schema. In addition, we described the several statistics that can be obtained via the SPARQL endpoint. There is a possibility that those statistics and other statistics are utilized for urban design.

In the future, we extend the methodology in order to design practical and ontologically consistent schema. Also, we will increase the amount of observation data and factors. Then, we will visualize a specified range of circles that indicate observation points in the tweet application. In addition, we will build polygon data of no-parking zones and ranges of IPBs as LOD. Also, we visualize the various aspects of the IPBs and design a schema that could be visualized easily. Furthermore, we consider collaborating with municipal governments and NPOs to collect more widespread data.

Acknowledgments

References

[1] N. Nishi, "The 2nd Health Japan 21: goals and challenges," The Journal of the Federation of American Societies for Experimental Biology, vol.28, no.1, 2014, 632.19.

[2] Ministry of Internal Affairs and Communications (2015), "Current bicycle usage and bicycle-related accident," http://www.soumu.go.jp/main content/000354710.pdf (accessed: September 10, 2015) (Japanese).

[3] A. Lawrynowicz and I. Lawniczak, "The Hazardous Situation Ontology Design Pattern," In: Proceedings of the 6th Workshop on Ontology and Semantic Web Patterns (WOP 2015) co-located with the 14th International Semantic Web Conference (ISWC 2015), 2015.

[4] A. Gangemi and P. Mika, "Understanding the semantic web through descriptions and situations," On The Move to Meaningful Internet Systems 2003: CoopIS, DOA, and ODBASE. Springer Berlin Heidelberg, 2003, pp.689-706.

[5] R. Segers., P. Vossen, M. Rospocher, L. Serafini, E. Laparra, G. Rigau, "Eso: A frame based ontology for events and implied situations," In: Proceedings of MAPLEX 2015, 2015.

[6] DP. Deng, GS. Mai, TR. Chuang, R. Lemmens, and KT. Shao, "Social Web Meets Sensor Web: From User-Generated Content to Linked Crowdsourced Observation Data," In: Proceedings of the Workshop on Linked Data on the Web co-located with the 23rd International World Wide Web Conference (WWW 2014), 2014.

[7] L. V. Ahn, "Games with a purpose," IEEE Computer, vol.39, no.6, 2006, pp. 92-94.

[8] M. Zook, M. Graham, T. Shelton, and S. Gorman, "Volunteered Geographic Information and Crowdsourcing Disaster Relief: A Case Study of the Haitian Earthquake," World Medical & Health Policy, vol.2, no.2, 2010, pp.7-33.

[9] I. Celino, D. Cerizza, S. Contessa, M. Corubolo, D. Dell'Aglio, E. D. Valle, S. Fumeo, and F. Piccinini, "Urbanopoly: Collection and Quality Assesment of Geo-spatial Linked Data via a Human Computation Game," In: Proceedings of the 10th Semantic Web Challange, 2012.

[10] S. Egami, T. Kawamura, and A. Ohsuga, "A Solution to Visualize Open Urban Data for Illegally Parked Bicycles" Transactions on Large-Scale Data and Knowledge-Centered Systems XXVII, 2016, pp.129-141.

[11] S. Egami, T. Kawamura, and A. Ohsuga, "Building Urban LOD for Solving Illegally Parked Bicycles" In: Proceedings of the 15th International Semantic Web Conference, 2016, pp.291-307.

[12] R. Mizoguchi, M. Ikeda, K. Seta, and J. Vanwelkenhuysen, "Ontology for Modeling the World from Problem Solving Perspectives," In: Proceedings of the 1995 International Joint Conference on AI (IJCAI) Workshop on Basic Ontological Issues in Knowledge Sharing, 1995, pp.1-12.

[13] S. Hwang and D. Yu, "GPS Localization Improvement of Smartphones UsingBuilt in Sensors," International Journal of Smart Home, vol.6, no.3, 2012, pp.1-8.

[14] C. Stadler, J. Lehmann, K. Höffner, and S. Auer, "LinkedGeoData: A Core for a Web of Spatial Open Data," Semantic Web Journal, vol.3, no.4, 2012, pp.333-354.

[15] M. Compton, P. Barnaghi, L. Bermudez, R. García-Castro, O. Corcho, S. Cox, J. Graybeal, M. Hauswirth, C. Henson, A. Herzog, V. Huang, K. Janowicz, W. D. Kelsey, D. L. Phuoc, L. Lefort, M. Leggieri, H. Neuhaus, A. Nikolov, K. Page, A. Passant, A. Sheth, K. Taylor, "The SSN Ontology of the W3C Semantic Sensor Network Incubator Group," Journal of Web Semantics: Science, Services and Agents on the World Wide Web, Vol.17, 2012, pp.25-32.

[16] S. Batsakis, E. Petrakis, Sowl, "A framework for handling spatio-temporal information in owl 2.0," Proceedings of the 5th Inter- national Symposium on Rules: Research Based and Industry Focused (RuleML), 2011, pp.242-249.

[17] C. Welty, R. Fikes, "A reusable ontology for fluents in owl," Frontiers in Artificial Intelligence and Applications Vol.150, 2006, pp.226-236.

[18] B. Harbelot, H. Arenas, C. Cruz, "Lc3: A spatio-temporal and semantic model for knowledge discovery from geospatial datasets," Journal of Web Semantics: Science, Services and Agents on the World Wide Web Vol.35, 2015, pp.3-24.