

# Question-Answering for Agricultural Open Data

Takahiro Kawamura<sup>1,2</sup>(✉) and Akihiko Ohsuga<sup>2</sup>

<sup>1</sup> Corporate Research & Development Center,  
Toshiba Corporation, Tokyo, Japan  
`takahiron.kawamura@toshiba.co.jp`

<sup>2</sup> Graduate School of Information Systems,  
University of Electro-Communications, Tokyo, Japan

**Abstract.** In the agricultural sector, the improvement of productivity and quality with respect to such attributes as safety, security and taste has been required in recent years. We aim to contribute to such improvement through the application of Information and Communication Technology (ICT). In this paper, we first propose a model of agricultural knowledge by Linked Open Data (LOD) with a view to establishing an open standard for agricultural data, allowing flexible schemas based on ontology alignment. We also present a semi-automatic mechanism that we developed to extract agricultural knowledge from the Web, which involves a bootstrapping method and dependency parsing, and confirmed a certain degree of accuracy. Moreover, we present a voice-controlled question-answering system that we developed for the LOD using triplification of query sentences and graph pattern matching of the triples. Finally, we confirm through a use case that users can obtain the necessary knowledge for several problems encountered in the agricultural workplace.

**Keywords:** Question-answering system · Linked Open Data · Agriculture

## 1 Introduction

Concern about food shortages affecting people in various parts of the world has been rising in recent years. However, since the expansion of the cultivated area is subject to constraints, it is necessary to improve productivity. On the other hand, the improvement of quality (safety, security, taste) is required in order to raise the incomes of farming households. In these circumstances, we aim to contribute to agricultural production by applying ICT techniques. Unlike producers involved in industrial production, most farmers, except for exemplary ones, are often confronted by unanticipated questions pertaining to several activities ranging from planting to harvesting, since agricultural work depends on complicated environmental factors. Thus, research[1] has been conducted with a view to developing a search engine to find previous problems similar to current problems. However, searching the Internet using a smartphone or a tablet PC on site

is disadvantageous in that the user must input keywords and iteratively tap and scroll through a Search Engine Result Page (SERP) to find an answer. Therefore, this paper proposes a voice-controlled question-answering system for search of agricultural information. Voice control is suitable for agricultural work since users typically have dirty hands and can speak freely without disturbing other people. At the same time, it provides a mechanism for registering the work of the user, since data logging is the basis of precision farming according to the Japanese Ministry of Agriculture.

There are many sources of agricultural information on the Web. There are also many databases (DBs) authorized by agricultural organizations. However, the sources on the Web are written in natural language and the DBs do not employ uniform schema, making it impossible to search them using standardized procedures. Moreover, the DBs may have open application programming interfaces (APIs) for search, but contents are closed in many cases. Therefore, we propose Linked Open Data (LOD) for agricultural knowledge with a view to establishing an open standard for agricultural data.

This remainder of this paper is organized as follows. Section 2 introduces related work regarding standardization of agricultural data and applications supporting the work. In section 3, we propose Plant Cultivation LOD and describe a mechanism of LOD extraction from the Web. Then, in section 4, we describe the development and evaluation of the question-answering system for agricultural work using LOD. Finally, we conclude by referring to future work in section 5.

## 2 Related Work

A standard for agricultural data, agroXML[2], which is an XML schema for describing agricultural work, has been proposed. It is used as a means of exchanging data in a structured and standardized way between farm and external stakeholders (government, manufacturer, and retailer) in the EU. In practice, only the necessary part of the schema is exchanged according to the purpose. However, elements such as “cultivation” and “WorkProcess” in agroXML are mainly for data logging, and not for the description of cultivation knowledge. agroXML also adopts a hierarchical XML schema, making it difficult to trace content such as graphs (although the graph version is under development).

Regarding semantic application of agricultural data, the Food and Agriculture Organization (FAO) of the United Nations[3] is currently developing the Agricultural Ontology Service Concept Server, whose purpose is the conversion of the current AGROVOC thesaurus to Web Ontology Language (OWL) ontologies. AGROVOC is a vocabulary containing 40000 concepts in 22 languages covering agricultural subject fields, and expressed in the W3C Simple Knowledge Organization System (SKOS) and also published as LOD. To the best of our knowledge, however, AGROVOC does not include knowledge of plant cultivation.

With regard to filed applications for agriculture, Fujitsu Ltd. offers a recording system that allows the user to simply register work types by buttons on

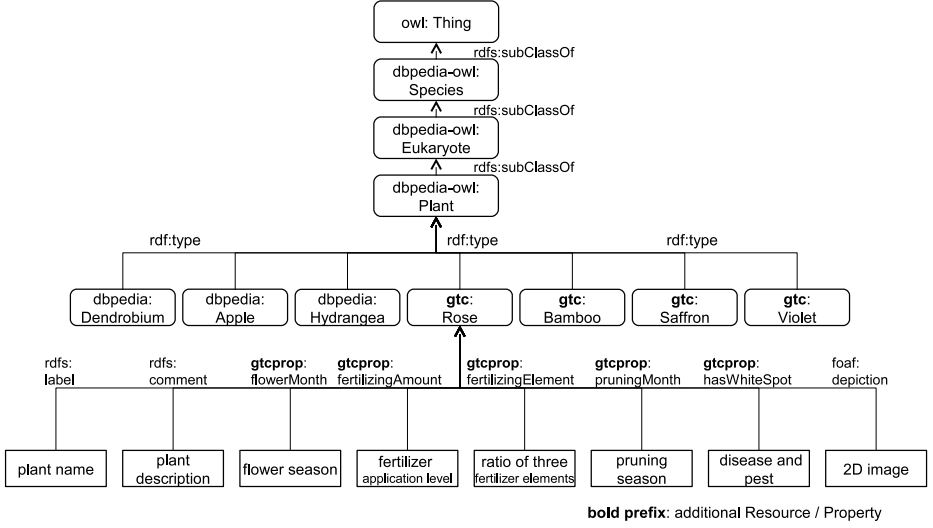


Fig. 1. Plant cultivation LOD

a screen with photos of the cultivated plants. NEC Corp. offers a machine-to-machine (M2M) service for visualizing sensor information and supporting farming diaries. Both systems address recording and visualization of the work, although our system is aimed at the search of cultivation knowledge on site by means of a voice-controlled question-answering system.

Recently, Apple’s Siri has drawn attention to the question-answering system. Siri offers a high-accuracy voice recognition function and correctly answers the question in the case that the information source is a well-structured website such as Wikipedia. However, extracting the information from unstructured web sites often fails and Siri returns the search engine results page, and then the user needs to tap URLs from the list. Thus, LOD is a promising source of the question-answering system. In fact, IBM’s Watson uses Linked Data as the internal information source in part [4, 5]. [6] serves as a useful reference for surveying other question-answering systems, some of which uses LOD as the knowledge source. Although our proposed system is related to a number of works, it is distinguished by accuracy improvement and data acquisition by the user feedback and registration mechanism described in section 4. Also, there is no similar work in terms of agricultural application.

### 3 Open Data for Agricultural Information

#### 3.1 Plant Cultivation LOD

Although the agroXML approach is ideal, adhering to a unique schema would pose difficulties for broader use. Therefore, we propose modeling cultivation

knowledge by LOD, which allows flexible schemas based on the premise of their alignment using ontologies such as OWL. Figure 1 presents an overview of the Plant Cultivation LOD, where each plant is an instance of the “Plant” class of DBpedia[7] to which we refer as a base. DBpedia is a Linked Open Data dataset extracted from part of Wikipedia content. DBpedia is well known throughout the world. By September 2011, DBpedia had grown to 31 billion Resource Description Framework (RDF) triples. DBpedia has already defined more than 10,000 plants as types of the Plant class and its subclasses such as “FloweringPlant”, “Moss” and “Fern”. In addition, we defined 104 plants, mainly species native to Japan. Each plant of the Plant class has almost 300 Properties, but most of them are biologically inherited from “Thing”, “Species” and “Eukaryote”. Thus, we added 67 properties to represent necessary attributes for plant cultivation. In terms of the LOD Schemas for logging the work, we prepared Properties mainly for recording dates of flowering, fertilizing, and harvesting. The LOD is written in RDF, and currently stored in a cloud DB, DYDRA, whose SPARQL (SPARQL Protocol and RDF Query Language) endpoint is open to the public. The following listing is a snippet of the LOD in N3 notation, where the first column indicates a property name, and the second column is a brief description of the property.

```
@prefix dbpedia-owl: <http://dbpedia.org/ontology/>
@prefix dbpprop: <http://dbpedia.org/property/>
@prefix gtcprop: <http://www.uec.ac.jp/gtc/property/>

<http://dbpedia.org/resource/basil> a dbpedia-owl:Plant;
rdfs:subClassOf depedia-owl:Eukaryote;
rdfs:label "Japanese name", "English name";
dbpprop:regionalOrigins "Asia";
rdfs:comment "Basil, or Sweet Basil, is a common name for
...";
foaf:page "reference page (url)";
foaf:depiction "picture (url)";
gtcprop:priceValue "market price";

# For cultivation knowledge
gtcprop:sunlight "degree of illuminance";
gtcprop:perennial 'true' | 'false';
gtcprop:difficulty "cultivation difficulty";
gtcprop:soil "type of soil";

gtcprop:lowestTemperature "MIN temperature for growth";
gtcprop:highestTemperature "MAX temperature for growth";
gtcprop:wateringAmount "degree of watering";
gtcprop:plantingMonth "start month for planting";
gtcprop:plantingMonthEnd "end month for planting";
gtcprop:flowerMonth "start month of blooming";
gtcprop:flowerMonthEnd "end month of blooming";
gtcprop:fertilizingAmount "degree of fertilization";
gtcprop:fertilizingMonth "season of fertilization";
gtcprop:fertilizingElement "chemical elements of fertilization";
gtcprop:pruningMonth "season for pruning";
gtcprop:pruningWay "method of pruning";
gtcprop:fruitMonth "season of harvesting";

# For disease and pest
gtcprop:hasWhiteSpot "possible reason for the case (wikipedia uri)";
gtcprop:hasBlackSpot "possible reason for the case (wikipedia uri)";
gtcprop:hasBrownSpot "possible reason for the case (wikipedia uri)";
```

```

gtcprop:hasYellowSpot "possible reason for the case (wikipedia uri)";
gtcprop:hasMosaic      "possible reason for the case (wikipedia uri)";
gtcprop:hasFade        "possible reason for the case (wikipedia uri)";
gtcprop:hasKnot        "possible reason for the case (wikipedia uri)";
gtcprop:hasMold        "possible reason for the case (wikipedia uri)";
gtcprop:hasInsect      "possible reason for the case (wikipedia uri)";
gtcprop:hasNoFlower    "possible reason for the case (wikipedia uri)";

# For work logging
gtcprop:plantingSpace   "indoor or outdoor";
gtcprop:plantingDateTime "date and hour of planting";
gtcprop:plantingAddress "address";
gtcprop:plantingWeather "weather";
gtcprop:plantingHighTemp "highest temperature of the day";
gtcprop:plantingLowTemp  "lowest temperature of the day";

gtcprop:flowerSpace     "indoor or outdoor";
gtcprop:flowerDateTime  "date and hour of blooming";
gtcprop:flowerAddress   "address";
gtcprop:flowerWeather   "weather";
gtcprop:flowerHighTemp  "highest temperature of the day";
gtcprop:flowerLowTemp   "lowest temperature of the day";

gtcprop:wateringSpace   "indoor or outdoor";
gtcprop:wateringDateTime "date and hour of watering";
gtcprop:wateringAddress "address";
gtcprop:wateringWeather "weather";
gtcprop:wateringHighTemp "highest temperature of the day";
gtcprop:wateringLowTemp  "lowest temperature of the day";

gtcprop:fertilizingSpace "indoor or outdoor";
gtcprop:fertilizingDateTime "date and hour of fertilization";
gtcprop:fertilizingAddress "address";
gtcprop:fertilizingWeather "weather";
gtcprop:fertilizingHighTemp "highest temperature of the day";
gtcprop:fertilizingLowTemp  "lowest temperature of the day";

gtcprop:purchaseSpace   "indoor or outdoor";
gtcprop:purchaseDateTime "date and hour of purchase";
gtcprop:purchaseAddress  "address";
gtcprop:purchaseWeather  "weather";
gtcprop:purchaseHighTemp "highest temperature of the day";
gtcprop:purchaseLowTemp  "lowest temperature of the day".

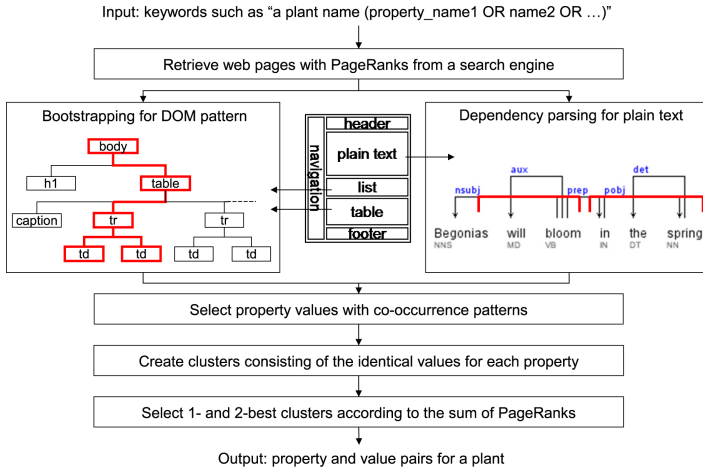
```

### 3.2 Agricultural Information Extraction from Web

This section describes a method for extracting cultivation knowledge from the Web, and constructing LOD. Our proposed method is inspired by [8] at AAAI10, which proposed a semi-automatic extraction service from the Web using the existing ontologies, where several learning methods are combined to reduce extraction errors. Although [8] focused on the world knowledge, and thus the granularity and the number of properties for each instance are rather abstract and limited, our method retains the variety of the properties and keeps the extraction accuracy by restricting the domain of interest.

**LOD Extraction Method.** We developed a semi-automatic method for growing the existing LOD to collect the necessary plant information from the Web and correlate it to DBpedia, which includes a dependency parsing method based

on WOM Scouter[9] and a bootstrapping method based on ONTOMO[10]. In the plant information, the plant names are easily collected from a list on any gardening web sites, and also we have already defined the property names from the aspect of the plant cultivation. We thus need the value of the property for each plant. The process of our LOD extraction is shown in Fig. 2.



**Fig. 2.** Process of LOD content generation

We first create a keyword list, which consists of an instance name, that is, plant name and a logical disjunction of the property names, such as “basil” (“Japanese name” OR “English name” OR “country of origin” OR ...), and then search on a web search engine, and receive more than 100 web pages. We then retrieve the page contents, except for PDF files and also take a Google PageRank value for each page.

The bootstrapping method extracts specific patterns of the document object model (DOM) tree in the page contents using some keys, which are the property names or their synonyms, and then applies the patterns to other web pages for the extraction of other property values. The method is used for the extraction from structured parts of the page contents like tables and lists.

There, however, are a number of gardening web sites, where most of the page contents are described in plain text. We thus developed an extraction method using dependency parsing, since a triple  $\langle \text{plantname}, \text{property}, \text{value} \rangle$  is regarded as a dependency relation  $\langle \text{subject}, \text{verb}, \text{object} \rangle$ . The method follows dependency relations in a sentence from a seed term, which is a plant name, a property name, or their synonym, and then extracts a triple, or a triple without a subject in the case of no subject within a sentence (the subject will be filled with a plant name in the keyword list later).

Next, we select property values that match with co-occurrence strings which are prepared for each property name, for example, the “temperature” property

must match with °C or °F. We then create clusters of the identical property values for each property based on Longest Common Substring (LCS), add up the PageRank values of the source web pages in each cluster, in order to excludes errors of the extraction and of the information source, then to determine the best possible property value and the second-best. Experienced gardeners finally select a correct value for each property from the extracted values. If there are various theories as to the correct value for the property, they selected the dominant one.

**LOD Extraction Accuracy.** The LOD extraction method was evaluated for 13 properties values of 90 plants. Table 1 shows precisions and recalls (avg.) of the best possible value (1-best) separated by the whole process, the bootstrapping method, and the dependency parsing. The precisions and recalls of the second-best possible value (2-best) of the whole process is also shown in the table. Although we retrieved more than 100 web pages for each plant, DOM parse errors and difference of file types reduced the page amount to about 60%. In the case that the sum of the PageRank values of two clusters are the same, two values are regarded as the first position. In addition, the accuracy is calculated in units of the cluster instead of each extracted value. In the case of 1-best, a cluster which has the biggest PageRank value is an answer for the property. In the case of 2-best, the two biggest clusters are compared with a correct value, and if either of the answers is correct, it is regarded as correct. N-best precision is defined as follows:

$$N - best\ precision = \frac{1}{|D_q|} \sum_{1 \leq k \leq N} r_k$$

,where  $|D_q|$  is the number of correct answers for a query  $q$ , and  $r_k$  is a function equaling 1 if the item at rank  $k$  is correct, zero otherwise.

The result of 1-best achieved a precision of 85% and a recall of 77%, and the 2-best achieved a precision of 97% and a recall of 87%. We thus confirmed that it is possible to present the binary choice including a correct answer in many cases. The automatic extraction will not be perfect after all, and then manual checking is necessary at the final step. Therefore, the binary choice is a realistic design. In more detail, the bootstrapping collected smaller amounts of values, and the recall was lower than the dependency parsing. However, the precision was higher than the dependency parsing. The reason is that data written in tables was correctly extracted, but lacks diversity of properties. The dependency parsing collected a large amount of values including many noisy data, and then the total accuracy was affected by the dependency parsing. The reason is that the biggest cluster of the PageRank value was composed of the values extracted by the dependency parsing. We thus plan to set some weights on the values extracted by the bootstrapping.

**Table 1.** Extraction accuracy (%)

Accuracy	1-best	2-best	1-best by bootstrapping	1-best by dependency parsing
Precision	85.2	97.4	88.6	85.2
Recall	76.9	87.2	46.2	76.9
Amount Ratio	—	—	10.8	89.2

## 4 Question-Answering System for Agriculture

### 4.1 Problem and Approach

The basic operation of our question-answering system is extraction of a triple such as *subject*, *verb*, and *object* from a query sentence by using morphological analysis and dependency parsing. Any question words (what, where, when, why, etc. are then replaced with a variable and the LOD DB is searched. In other words, the  $\langle \textit{subject}, \textit{verb}, \textit{object} \rangle$  triples in the LOD DB are matched against  $\langle ?, \textit{verb}, \textit{object} \rangle$ ,  $\langle \textit{subject}, ?, \textit{object} \rangle$ , and  $\langle \textit{subject}, \textit{verb}, ? \rangle$  in the query. SPARQL is based on graph pattern matching, and this method corresponds to a basic graph pattern (one triple matching). At the data registration, if there is a resource corresponding to the *subject* and a property corresponding to the *verb* from the user statement, a triple that has the *object* from the user statement as the value is added to the DB.

However, since the schema is open, mapping of query sentence to the schema poses a problem. Although mapping between the verb in the query sentence (in Japanese) and a Property in the LOD schema of the DB must be defined in advance, both of them are unknown in this open schema scenario (in the closed DB the schema is given), so the score according to the mapping degree cannot be predefined. The open schema means that the schema is not regulated by any organization, and there may be several properties of the same meaning and a sudden addition of a new property. In addition, we assume searching over multiple LOD sets made by the different authors. We therefore use a string similarity and a semantic similarity technique using the WordNet thesaurus from the field of ontology alignment to map verbs to Properties, and attempt to improve the mapping based on user feedback. We first register a certain set of mappings  $\{\textit{verb}, \textit{property}\}$  as seeds in the Key-Value Store (KVS). If a verb is unregistered, we then do the following:

- (1) Expand the *verb* to its synonyms using Japanese WordNet ontologies, and then calculate the LCS with the registered *verbs* to use as the similarity.
- (2) Translate the new *verb* into English, and calculate the LCS of the English with the registered properties.
- (3) If we find a resource that corresponds to a *subject* in the query sentence in the LOD, we then calculate the LCSs of the translated *verbs* with all the properties belonging to the resource, and create a ranking of possible mappings according to the combination of the above LCS values.



- (4) The user feedback that indicates which property was actually viewed is sent to the server, and the corresponding mapping of the new *verb* to the property is registered in the KVS.
- (5) Since the registered mappings are not necessarily correct, we recalculate the confidence value of the mapping based on the number of pieces of feedback, and update the ranking of the mapping to improve the N-best accuracy (see section 4.3).

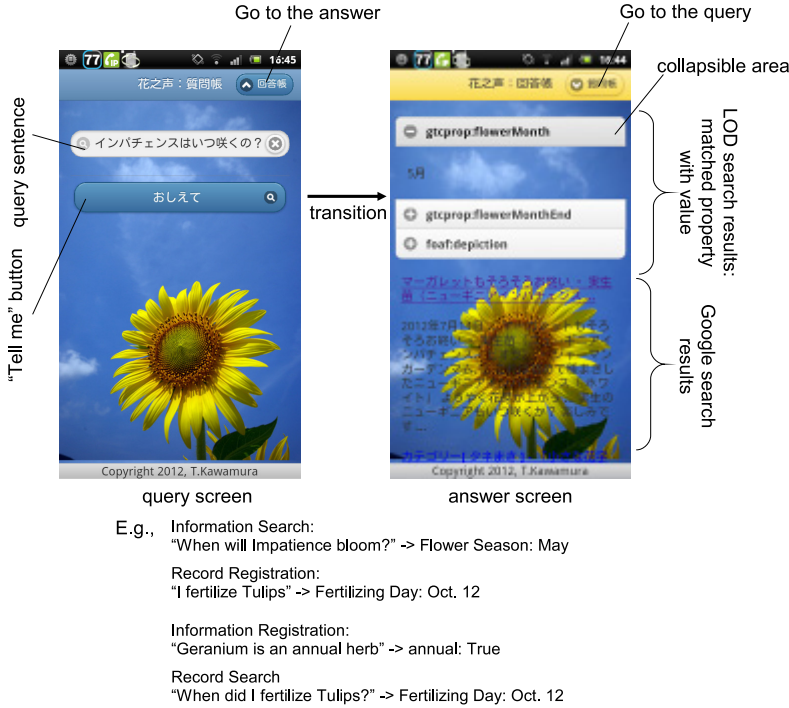
Also, we provide a registration mechanism of the user context information to increase contents of the Plant Cultivation LOD. When the user registers a sentence in the DB, the sensor data are automatically aggregated by using built-in sensors on the smartphone, and the context information at that time and location is inserted into the DB. For example, when a user registers a triple describing a flower has blossomed, the sensor data for the location is converted to literal, one for the temperature is converted to integer, and one for the space is translated to Indoor or Outdoor, respectively. Then, the context information such as `gtcprop:flowerAddress` (location), `gtcprop:flowerDateHighTemp` (highest temperature of the day), `gtcprop:flowerDateLowTemp` (lowest temperature of the day), and `gtcprop:flowerSpace` (space of the flower) is automatically registered in the LOD DB. We prepared the LOD schemas (properties) corresponding to the context information (see ‘work logging’ part of the previous listing). Therefore, the user can register not only the direct assertion, but also several background information at once.

## 4.2 Development of Question-Answering System

Figure 3 shows an interface of our question-answering system. It automatically classifies the speech intention (Question Type) of the user into the following four types (Answer Type is a literal, URI, or image).

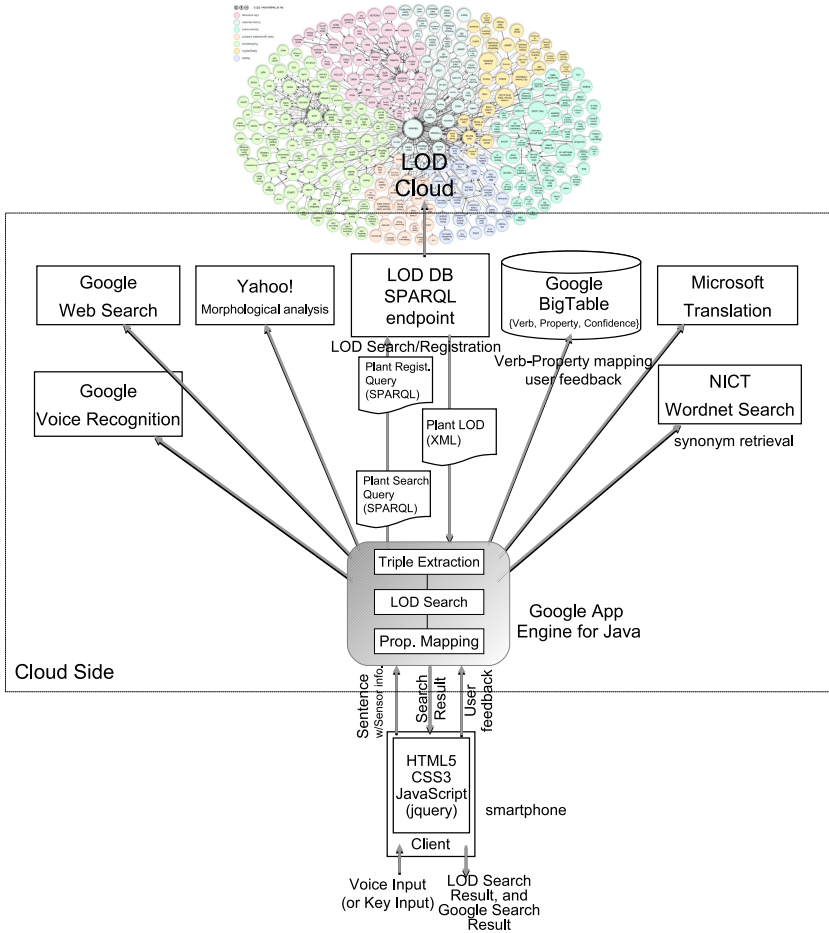
1. Information Search  
Search for plant information in the LOD DB.
2. Information Registration  
Register new information for a plant that does not currently exist in the LOD DB or add information to an existing plant.
3. Record Registration  
Register and share records of the daily work. However, the verbs that can be registered are limited to the predefined Properties in the LOD.
4. Record Search  
Search through records to review previous work and view the work of other people.

Figure 4 shows the architecture of our question-answering system. The user can input a query sentence by Google voice recognition or keyboard. The system then accesses the Yahoo! API for Japanese morphological analysis, extracts a triple using the built-in dependency parser, and generates a SPARQL query by filling in slots in a query template. The search results are received in XML format.



**Fig. 3.** Interface of QA system

After searching the {verb, property} mappings registered in Google Big Table and accessing the Microsoft Translator API and Japanese WordNet Ontology provided by the National Institute of Information and Communications Technology (NICT), the LCS values for each mapping are calculated as described in section 4.1. The order of matching is firstly matching the Subject against Resources by tracing ‘sameAs’ and ‘wikiPageRedirects’ links, and then searching for Verb matches with the Properties of the Resources. A list of possible answers is then created from the pairs of Properties and Values with the highest LCS values. The number of answers in the list is set to three owing to constraints on the client UI. The results of a Google search are also shown below in the client to clarify the advantages and limitations of the QA system by comparison. The user feedback is obtained by opening and closing a collapsible area in the client that gives a detailed look at the Value of the Property. During searches, the feedback updates the confidence value of a registered mapping {verb, property} or registers a new mapping. During registration, the feedback has the role of indicating which of three properties the object (value) should be registered to. The client UI displays the results. Text-to-speech has not been implemented yet. The query currently matches the graph pattern  $\langle \text{subject}, \text{verb}, ? \rangle$  only. As the target LOD, the QA system can search not only on the Plant Cultivation LOD, but also on DBpedia.



**Fig. 4.** Mashup architecture

The context registration mechanism is realized by the acquisition of sensor data. The sensor data are obtained by Android OS 2.2+ functions and jQuery 1.6.4+ running on the smartphone, and the related web services. For example, date and time information are obtained from the internal clock, and location is obtained by the Global Positioning System (GPS) function in the Android OS. But Point of Interest (POI) and weather information (temperature and humidity) are obtained by accessing Yahoo! Open Local Platform (olp.yahoo.co.jp), and Japan Meteorological Agency (www.jma.go.jp) based on the time and the GPS information. The POI specifies location names (buildings, companies, stations) around that location. Furthermore, we determine the location as indoor or outdoor space by the built-in illuminance sensor.

Our system is available at [www.ohsuga.is.uec.ac.jp/~kawamura/fv.html](http://www.ohsuga.is.uec.ac.jp/~kawamura/fv.html) (in Japanese) and won a Judges's Special Award in the LOD Challenge Japan 2012.

### 4.3 Evaluation in a Use Case

In this section, we set a specific use case and conduct an evaluation on the accuracy improvement and the context registration.

**Accuracy Improvement.** As a use case where the question-answering system is useful, we focused on a situation where the user asks the reason for the problems of plant growth such as physiological disorder, disease and pest. We collected questions on plant growth from Q&A sites such as Yahoo! Answers, and translated some of them to the spoken language as a dataset. Examples are as follows: Why have the leaves of black pine died? / Does corn need fertilizer? / Where is a suitable space for sage? / Why has the bark of apple become brown? / Does leaf curl affect passion fruit? / Why have some edges of the leaves of kalanchoe been dying? / Is tomato with white patterns safe to eat? However, we have limited the questions to those for the plant species registered in the LOD. In the experiment, we divided 90 collected questions into 10 sets. Then, we randomly selected and evaluated the first set and the next set consecutively as a test. We gave the correct feedback (the correct answer was also retrieved from the Q&A sites), which means the registration of {verb, property} mapping and incrementation of the confidence value, to one of the three answers per query. After the evaluation of the second set, we cleared all the effect of the user feedback, and repeated the above from the first set. The difference of the accuracy between the first and the second corresponds to the improvement of the user feedback. We assumed that the query sentence is correctly entered and did not consider voice recognition error, since we can select the correct sentence from the results of the Google voice recognition. The result is shown in table 2.

**Table 2.** Accuracy of search

	False		True	
	no Prop.	triplication error	1-best	3-best
1st Set (avg.)	22.2%	11.1%	55.6%	66.7%
2nd Set (avg.)			66.7%	66.7%

In the table, the result of False shows the average of the first set and the second set. We found that the coverage of the prepared Properties remains approx. 80% of the questions. We plan to expand the Properties defined in the Plant Cultivation LOD. In addition, the accuracy of the conversion from a sentence to a triple (triplication) was rather high, almost 90%. The current extraction mechanism is rule-based, but we intend to extend the rules and use of machine learning techniques to manage the broader questions. On the other hand, the result of True was about 67% of the accuracy in 3-best. By comparing the 1-best result for the first set with the second one, we can confirm that the problem raised in section 4.1, the mapping of the verb in the question to the LOD schema, has been improved about 10% by the user feedback. (Note that 1-best accuracy

equals 3-best accuracy means all the correct answers are in the first position, that is, they are amongst the first three positions.) However, this use case only focuses on the questions related to ill-growth of the plant. Moreover, we evaluated the search function in this experiment, but the accuracy of the registration has not been studied yet. In the near future, we intend to collect a broader range of questions and registration statements, and conduct the evaluation on the accuracy and the performance.

**Context Registration.** We also conducted experiments to confirm the usefulness of the context registration. In the experiment, we also used the dataset regarding questions and answers on plant growth retrieved from the Q&A sites. Table 3 presents the properties representing a series of planting  $\rightarrow$  flowering  $\rightarrow$  pruning  $\rightarrow$  withering that are registered by the user statement, and the context information obtained by the sensors. In the table, the intersection of the line of ‘gtcprop:flowerDate’ and the column of ‘Location’ means that when a user registers the date a flower has blossomed, then the Location information is automatically registered at the same time. Then, the number where the line and the column intersect means the co-occurrence ratio of these two information in the answers of the dataset. That is, answers for questions about the time when a flower blossoms include the Location information with 54.5% probability in the dataset. The size of the dataset is as follows: 414 answers for ‘plantingDate’, 114 answers for ‘flowerDate’, 128 answers for ‘pruningDate’, and 99 answers for ‘dieDate’. But we excluded short answers consisting of only one or two lines. In this experiment, we considered that the additional information that has been described together with the original information in the answer is worth registering to the Plant Cultivation LOD. Therefore, we regarded that the automatic registration of such context information has usefulness.

**Table 3.** Co-occurrence ratio (%) of registered property and collected context

registered property	Time	Space {Indoor, Outdoor}	Location	Weather	High Temp.	Low Temp.	Humid
gtcprop:plantingDate	0.0	42.9	14.3	28.6	42.9	57.1	42.9
gtcprop:flowerDate	9.1	45.5	54.5	18.2	36.4	45.5	27.3
gtcprop:pruningDate	0.0	11.1	11.1	0.0	0.0	11.1	11.1
gtcprop:dieDate	0.0	12.5	25.0	25.0	25.0	37.5	37.5
(avg.)	2.3	28.0	26.2	17.9	26.1	37.8	29.7

As a result, the context information has the variation in its usefulness (0.0% – 57.0%). However, most of them have at least more than 20% on the average, thus the context registration mechanism can be regarded useful.

## 5 Conclusion and Future Work

This paper proposed the Plant Cultivation LOD with a view to establishing an open standard for agricultural data that models cultivation knowledge with flexible schemas without adhering to a hierarchical structure, and then presented a mechanism that we developed to extract the necessary knowledge from the Web and an evaluation of its accuracy. Moreover, we proposed a voice-controlled question-answering mechanism for this open schema LOD in order to obtain knowledge of the problems that farmers encounter on site, and then evaluated its accuracy through a use case. We are now planning to expand the Plant Cultivation LOD, and also considering conducting an evaluation of our question-answering system in the agricultural workplace.

## References

1. Ninomiya, S.: ICT-enabled Agricultural Science for Development Scenarios, Opportunities, Issues. In: Proc. of Science Forum 2009, pp. 1-5 (2009)
2. Schmitz, M., Martini, D., Kunisch, M., Mosinger, H.J.: agroXML - Enabling Standardized, Platform-Independent Internet Data Exchange in Farm Management Information Systems. In: Metadata and Semantics, pp. 463-468 (2009)
3. Agricultural Information Management Standards. <http://aims.fao.org/> (accessed: September 25, 2014)
4. Welty, C., Barker, K., Aroyo, L., Arora, S.: Query Driven Hypothesis Generation for Answering Queries over NLP Graphs. In: Cudré-Mauroux, P., Heflin, J., Sirin, E., Tudorache, T., Euzenat, J., Hauswirth, M., Parreira, J.X., Hendler, J., Schreiber, G., Bernstein, A., Blomqvist, E. (eds.) ISWC 2012, Part II. LNCS, vol. 7650, pp. 228-242. Springer, Heidelberg (2012)
5. Welty, C., Murdock, J.W., Kalyanpur, A., Fan, J.: A Comparison of Hard Filters and Soft Evidence for Answer Typing in Watson. In: Cudré-Mauroux, P., Heflin, J., Sirin, E., Tudorache, T., Euzenat, J., Hauswirth, M., Parreira, J.X., Hendler, J., Schreiber, G., Bernstein, A., Blomqvist, E. (eds.) ISWC 2012, Part II. LNCS, vol. 7650, pp. 243-256. Springer, Heidelberg (2012)
6. Lopez, V., Uren, V., Sabou, M., Motta, E.: Is question answering fit for the semantic web? A survey Semantic Web J. **2**(2), 125-155 (2011)
7. Auer, S., Bizer, C., Kobilarov, G., Lehmann, J., Cyganiak, R., Ives, Z.G.: DBpedia: A Nucleus for a Web of Open Data. In: Aberer, K., Choi, K.-S., Noy, N., Allemang, D., Lee, K.-I., Nixon, L.J.B., Golbeck, J., Mika, P., Maynard, D., Mizoguchi, R., Schreiber, G., Cudré-Mauroux, P. (eds.) ASWC 2007 and ISWC 2007. LNCS, vol. 4825, pp. 722-735. Springer, Heidelberg (2007)
8. Carlson, A., Betteridge, J., Kisiel, B., Settles, B., Hruschka, Jr., E.R., Mitchell, T.M.: Toward an Architecture for Never-Ending Language Learning. In: Proc. of 25th Conference on Artificial Intelligence (AAAI), pp. 1306-1313 (2010)
9. Kawamura, T., Nagano, S., Inaba, M., Mizoguchi, Y.: Mobile Service for Reputation Extraction from Weblogs - Public Experiment and Evaluation. In: Proc. of 22nd Conference on Artificial Intelligence (AAAI), pp. 1365-1370 (2007)
10. Kawamura, T., Shin, I., Nakagawa, H., Nakayama, K., Tahara, Y., Ohsuga, A.: ONTOMO: Web Service for Ontology Building - Evaluation of Ontology Recommendation using Named Entity Extraction. In: Proc. of International Conference WWW/INTERNET 2010 (ICWI), pp. 101-111 (2010)

Transactions on Large-Scale Data- and

Knowledge-Centered Systems XVI

Selected Papers from ACOMP 2013

Hameurlain, A.; Küng, J.; Wagner, R.; Dang, T.K.; Thoai,  
N. (Eds.)

2014, IX, 103 p. 35 illus., Softcover

ISBN: 978-3-662-45946-1