Saudi Computer Society, King Saud University

Applied Computing and Informatics

(http://computer.org.sa)
www.ksu.edu.sa
www.sciencedirect.com

Applied Computing
& Informatics

ORIGINAL ARTICLE

Algorithms to analyze the quality test parameter values of seafood in the proposed ontology based seafood quality analyzer and miner (ONTO SQAM) model

Vinu Sherimon

Higher College of Technology, Muscat, Oman

Received 7 June 2016; revised 13 August 2016; accepted 14 August 2016
Available online 23 August 2016

KEYWORDS

Data store update;
Ontology;
Knowledge base;
Seafood;
Quality

Abstract Ensuring the quality of food, particularly seafood has increasingly become an important issue nowadays. Quality Management Systems empower any organization to identify, measure, control and improve the quality of the products manufactured that will eventually lead to improved business performance. With the advent of new technologies, now intelligent systems are being developed. To ensure the quality of seafood, an ontology based seafood quality analyzer and miner (ONTO SQAM) model is proposed. The knowledge is represented using ontology. The domain concepts are defined using ontology. This paper presents the initial part of the proposed model – the analysis of quality test parameter values. Two algorithms are proposed to do the analysis – Comparison Algorithm and Data Store Updater algorithm. The algorithms ensure that the values of various quality tests are in the acceptable range. The real data sets taken from different seafood companies in Kerala, India, and validated by the Marine Product Export Development Authority of India (MPEDA) are used for the experiments. The performance of the algorithms is evaluated using standard performance metrics such as precision, recall, and accuracy. The results obtained show that all the three measures achieved good results.

1. Introduction

In seafood processing industries, control measures are adopted to ensure the safety and quality of products. Various quality assurance standards exist to guarantee the safety and quality of goods. Some of them are International Organization for Standardization (ISO), Hazard Analysis Critical Control Point (HACCP), Quality Management Program (QMP), Total Quality Management (TQM), etc. [3]. So the adoption of

E-mail address: vinusherimon@yahoo.com

Peer review under responsibility of King Saud University.

ELSEVIER Production and hosting by Elsevier

http://dx.doi.org/10.1016/j.aci.2016.08.001

proper quality control systems is mandatory in any seafood organization that deals with food processing and distribution. Seafood exporters in developing countries have experienced serious problems complying with progressively stricter food safety and quality requirements for importers. So the author proposed ONTO SQAM, a knowledge-based model to ensure the quality of the seafood.

Knowledge-Based Systems are expert systems that utilize Artificial Intelligence (AI) techniques to generate knowledge from the data, analyze the data and convert it into meaningful information, thereby acting as a human expert. Ontologies propose an efficient knowledge representation technique in Artificial Intelligence systems. Ontologies provide sound reasoning systems. The definition of ontology follows as a "*formal, explicit specification of a shared conceptualization*" [1]. Ontologies were developed in AI to facilitate knowledge sharing and reuse [2]. Ontologies provide a formal semantics for the description of many concepts involved in the domain and relations between these concepts [6]. The knowledge base of the proposed model contains information about the seafood, details of the test parameters, values, etc. It is implemented using different ontologies. The ontologies analyze the quality test results, and a better, and detailed quality report is generated automatically by the system.

Every seafood organization is conducting organoleptic, microbiological, and chemical checks at all stages of seafood production. The standard values are used to compare the results of all the above tests, and the management takes appropriate measures when the quality is found to be poor. Currently, the companies do these procedures manually. But there is a possibility that sometimes the test values obtained will not be in the acceptable range. It happens due to several reasons such as test equipment failure, insufficient incubation and input errors. Currently, in the seafood industry, no electronic checks are done to figure out these types of mistakes that can happen at the initial stage. Also, the laboratory technologist is not comparing the test value with the values obtained in the past for similar trials. So in the proposed model, two algorithms are suggested to do the initial analysis of quality test parameter values. The aim of this is to understand about any errors or quality issues at the beginning stage itself. A comparison algorithm is proposed to ensure the validity of the test procedures and the accuracy of the values obtained. Data store update algorithm updates the data store with a minimum acceptable value and maximum value for each test. If there is a significant difference between the values currently generated and the previous values, the system alerts the technologists to check the test procedure or to repeat the tests again. In the proposed ONTO SQAM model, different algorithms are proposed for analysis, mining, and prediction of data. This paper presents only the initial part of the proposed model – the analysis of quality test parameter values.

The rest of the paper is organized as follows: Section 2 comprises of the materials and methods. The section describes the design of ontologies, analysis of quality test parameter values, and the proposed algorithms. Section 3 includes results and discussions. It illustrates the data sets used, the performance of the algorithms on the selected data sets and the values of performance metrics. Section 4 presents the conclusion which is followed by references.

2. Materials and methods

2.1. Knowledge modeling - design of ontologies

Ontologies define the semantics of the concepts identified in a domain. There is no single method to develop an ontology. This research follows the approach proposed by Noy and McGuinness in the development of ontologies [4]. Many interesting ontologies are available on the web in different domains, particularly in the medical domain. Such ontologies enable information sharing, and other applications reuse them. But in seafood domain, no relevant ontologies exist. So the ontologies used in the proposed model are developed from scratch.

The proposed ONTO-SQAM model comprises of three ontologies: the *Seafood* ontology, *Test Specifications* ontology, and *Seafood Integrated* Ontology. Fig. 1 presents the taxonomy of *Seafood* ontology.

The parent classes in *Seafood* ontology are the central concepts in the domain such as product types, fish categories, country of export and type of tests. *Country, Fish, Product, Test*, etc., are the main classes of the ontology. Each of these main classes consists of different subclasses according to their types. The main fish categories included in the ontology are *Bony Fish, Cephalopods, Crustaceans, Molluscs,* and *Scombridae*. The various seafood under each category is the instances of the particular class. For example, the instances of *Cephalopods* are *Cuttlefish, Octopus, Squid*, etc. Fig. 2 shows *Cephalopods* class and its individuals (instances).

Similarly, the *Test* subclass consists of subclasses such as *chemical*, *microbiological* and *organoleptic*. Each of these classes, in turn, consists of further subclasses as given in Fig. 1.

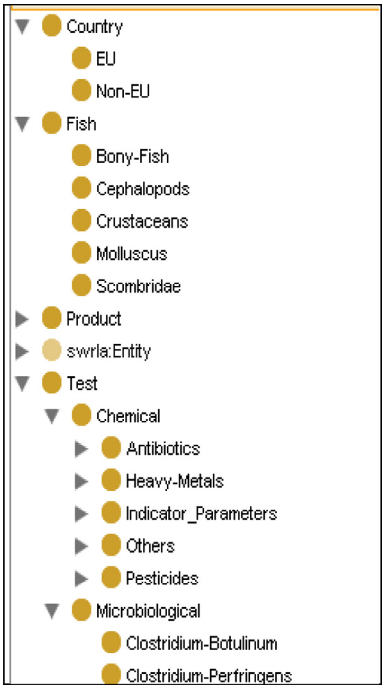
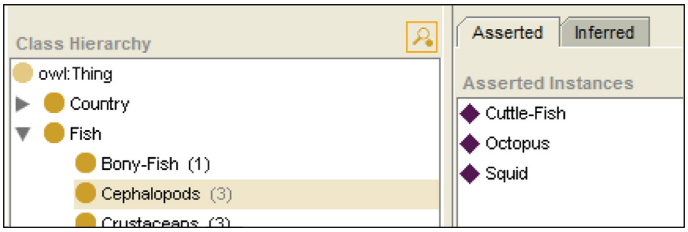


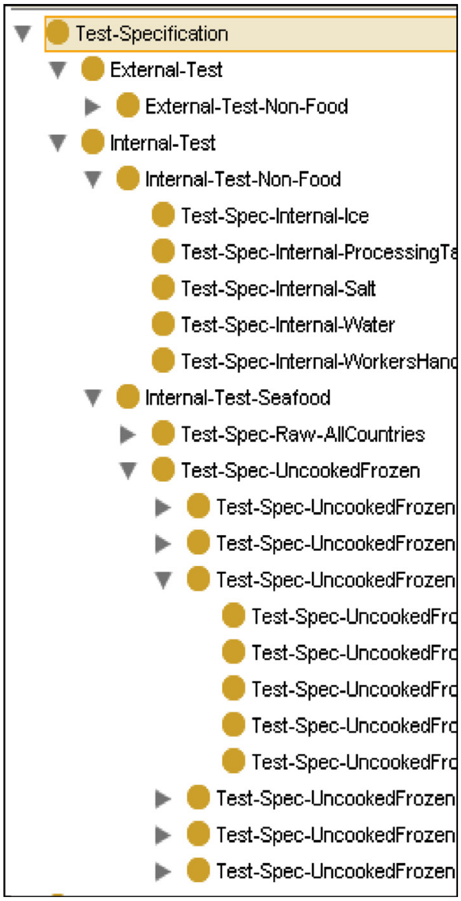
Figure 1 Taxonomy of seafood ontology.



- owl:Thing
  - Country
  - Fish
    - Bony-Fish (1)
    - Cephalopods (3)
    - Crustaceans (3)

- Cuttle-Fish
- Octopus
- Squid

Figure 2 Instances of *Cephalopods* subclass.



- Test-Specification
  - External-Test
    - External-Test-Non-Food
  - Internal-Test
    - Internal-Test-Non-Food
      - Test-Spec-Internal-Ice
      - Test-Spec-Internal-ProcessingTa
      - Test-Spec-Internal-Salt
      - Test-Spec-Internal-Water
      - Test-Spec-Internal-WorkersHand
    - Internal-Test-Seafood
      - Test-Spec-Raw-AllCountries
      - Test-Spec-UncookedFrozen
        - Test-Spec-UncookedFrozen
        - Test-Spec-UncookedFrozen
        - Test-Spec-UncookedFrozen
          - Test-Spec-UncookedFro
          - Test-Spec-UncookedFro
          - Test-Spec-UncookedFro
          - Test-Spec-UncookedFro
          - Test-Spec-UncookedFro
        - Test-Spec-UncookedFrozen
        - Test-Spec-UncookedFrozen
        - Test-Spec-UncookedFrozen

Figure 3 Taxonomy of test specifications ontology.

Fig. 3 shows the taxonomy of *Test Specifications* Ontology. It contains the test standards of seafood and sanitation items such as water, ice and salt used during the seafood processing.

Quality Checking tests are of two types – Internal Test and External Test. The specifications are different for internal tests and external tests. *External-Test* and *Internal-Test* are the two sub-classes of the *Test-Specification* ontology. Under each of these classes, we have subclasses to represent sanitation item and seafood item. *Internal-Test-Non-Food* and *Internal-Test-Seafood* are the two subclasses of *Internal-Test class.*

Test-Spec-Raw-AllCountries and *Test-Spec-Uncooked-Frozen* are the two primary classes of *Internal-Test-Seafood* class. For each category of raw seafood, the test specifications are independent of the country of export. So the five subclasses of *Test-Spec-Raw-AllCountries* represent the test specifications of five categories of seafood included in *Seafood* ontology. But in the case of frozen seafood, each country of export has its standards for specific seafood. So a subclass is defined for each country under the class *Test-Spec-UncookedFrozen* and for each of those classes, subclasses are defined to include the test standards of particular seafood.

Seafood Integrated Ontology is a storehouse of each product test file with the corresponding test report. This ontology is used by the proposed Data Store Update algorithm to update the test values as explained in the later sections.

2.2. Analysis of quality test parameter values – proposed model

Fig. 4 shows the proposed model to analyze the test parameter values. Initially, the laboratory technologist conducts tests (organoleptic, microbiological and chemical) as usual, and the sample results are input to the proposed model. The model generates a product test file (*owl* file) that contains the test parameters and the results of each sample. It produces a new file for each test sample, which contains the test parameters and the obtained parameter values.

Before checking the values with the test standards stored in the *Test Specifications* ontology, the comparison algorithm (Analysis I) analyzes the parameter values. This checking does ensure that the obtained values are in the acceptable range. Data store update algorithm updates the acceptable values for a particular test in the data store. The permitted value for each test item is computed by the data store update algorithm using ontology data and *Test Specifications* ontology. These values are stored in the Data Store, making it available for fast retrieval. The comparison algorithm compares the current test values with the values retrieved from the data store. If the values are not in the permissible range, the laboratory technologists provide necessary alerts.

If the values are in the acceptable range, the product test file is validated (Analysis II) as per the specifications stored in the *Test Specifications* ontology. Table 1 shows the microbiological, organoleptic and antibiotic test parameters with their standards to check the quality of molluscs raw material.

The test quality standards for validation are implemented using Semantic Web Rule Language (SWRL) [7]. Fig. 5 shows the rule in SWRL to validate the Molluscs. The microbiological tests check the presence of 'TPC', 'Vibrio Cholera', 'Vibrio-Parahaemolyticus', 'E-Coli' and 'Staphylococcus-Aureus' in molluscs. Organoleptic factors are ten in number, listed from 6 through 15 in Table 1. Paralytic shellfish poisoning (PSP) and Diarrhetic Shellfish Poisoning (DSP) are two major biotoxins associated with shellfishes such as clams, mussels and oysters. These 19 parameters determine the overall quality of the raw material. The technologist compares the values obtained with the tolerance limit. The acceptance/rejection of

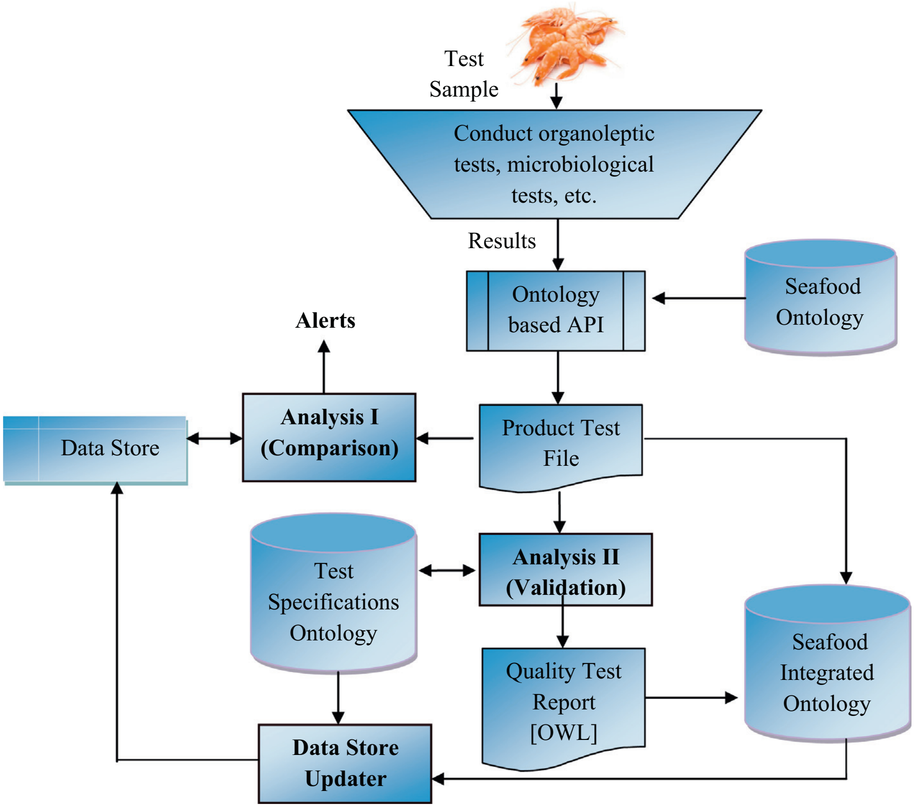


Figure 4 Analysis of quality test parameter values.

raw material is taken based on the comparison. A comprehensive quality report is generated describing the quality of the product tested. This report tells the acceptance status of the test input sample.

2.3. Proposed algorithms to analyze the test parameter values

The *Test Specifications* ontology stores the microbiological, chemical and organoleptic test standards (*min_value* and *max_value*). So these test standards are available only during the Analysis phase. Also due to scalability issues of ontology, it is time-consuming to every time retrieve the test standards directly from the ontology. So a data store update algorithm is proposed to compute the permitted value for each test item. The above algorithm stores the updated *min_value* and *max_value* of each test in the data store, making it available for fast retrieval. When the laboratory technologist inputs the value, a comparison algorithm retrieves the permitted range of the concerned test from the data store. The comparison algorithm checks whether the value entered by the user is within the allowable range, by comparing the test input value with *min_value* and *max_value* available in the data store.

2.3.1. Data store update algorithm

Table 2 shows the proposed Data Store Update algorithm. The purpose of this algorithm is to update the data store with an acceptable minimum and maximum value for each test. In the beginning, the algorithm retrieves the corresponding past test values from the ontology. Then the algorithm removes the duplicates and sorts the values. The next step is to set a minimum value and a maximum value. The *Test Specifications* ontology outputs the standard test limits (*min* and *max*) of the concerned test. The algorithm compares the minimum of the past test values with the standard minimum limit retrieved from the ontology and accordingly sets the *min_value*. Similarly, it sets the *max_value*. The algorithm also computes the variance of the test values. In the end, it updates the computed values *min_value*, *max_value*, and the *variance* in the data store. This algorithm is executed periodically as per the system settings so that the data store contains the updated values.

2.3.2. Comparison algorithm

Table 3 shows the proposed comparison algorithm. The purpose of this algorithm is to ensure the validity of the test procedures and the accuracy of the values obtained. So this comparison is not made with the standard limits of the test. Instead, the comparison algorithm checks the test values with the values in the data store. The algorithm retrieves the *min_value* and *max_value* of the current test from the data store and compares with the *current_value*. If the *current_value* is within the permissible range, it means that the values are known. Otherwise, the algorithm checks the range of variation of the test values from the data store values. For this, it computes the range and standard deviation. The algorithm retrieves the variance stored in the data store and calculates the standard deviation. Next step is to identify the severity of the error. If the *current_value* is lesser than the *min_value*, the algorithm checks whether it is also smaller than *min_value – std_dev*. If so, the algorithm concludes that the output is at the acceptable level. Otherwise, if the *current_value*

Table 1 Test specifications – raw material – molluscs.

Sl. no.	Parameters (microbiological, organoleptic and chemical)	Tolerance limit/ sample
1	Vibrio-Cholerae	Absent
2	Vibrio-Parahaemolyticus	MIN 0 and MAX 2.9
3	TPC	MIN 0 and MAX 5,00,000
4	E-Coli	MIN 0 and MAX 20
5	Staphylococcus-Aureus	MIN 0 and MAX 100
6	Damaged-Pieces	MIN 0, MAX 10% [by count]
7	Dehydration	MIN 0, MAX 10% [by count]
8	Deterioration	MIN 0, MAX 10% [by count]
9	Discoloration	MIN 0, MAX 10% [by count]
10	Foreign-Vegetable-Matter	1 or 2
11	General-Appearance	Bright
12	General-Odor	Good
13	Objectionable-Foreign-Matter	NIL
14	Non-Uniformity	NIL
15	Texture	Firm and soft
16	PSP	MIN 0, MAX 0.8 PPM
17	DSP	MIN 0, MAX 0.2 PPM
18	ASP	NOT DETECTED
19	NSP	NOT DETECTED

is lesser than (*min_value* – *range*), the algorithm alerts the technologists that there is a chance of error. If both the above conditions are false, it means that the *current_value* is not acceptable, and the algorithm generates the concerned alerts. Correspondingly the technologists can check the test values or can repeat the test. Similarly, if the *current_value* is greater than the *max_value*, the algorithm checks further as mentioned above and generates concerned alerts. Once we have a large number of test values and their reports in the data store, the comparison algorithm gets a collection of test values so that it can set the *min_value* and *max_value*. So the accuracy of the comparison algorithm is increased as more data accumulate in the data store.

3. Results and discussion

3.1. Datasets

The data sets used are taken from different seafood companies in Kerala, India, which exports seafood and seafood products to all around the world. The Marine Product Export Development Authority of India (MPEDA) validated the data sets. The data sets consist of details of seafood purchased from various landing centers. It includes the purchase date, seafood type, landing center code, quantity, importer details, microbiological, chemical, organoleptic test details and details of seafood exports to different countries. The experiment used three datasets of seafood families namely, Bony Fish, Cephalopods, and Molluscs. Each family includes different types of seafood, and each group contains valid and invalid data.

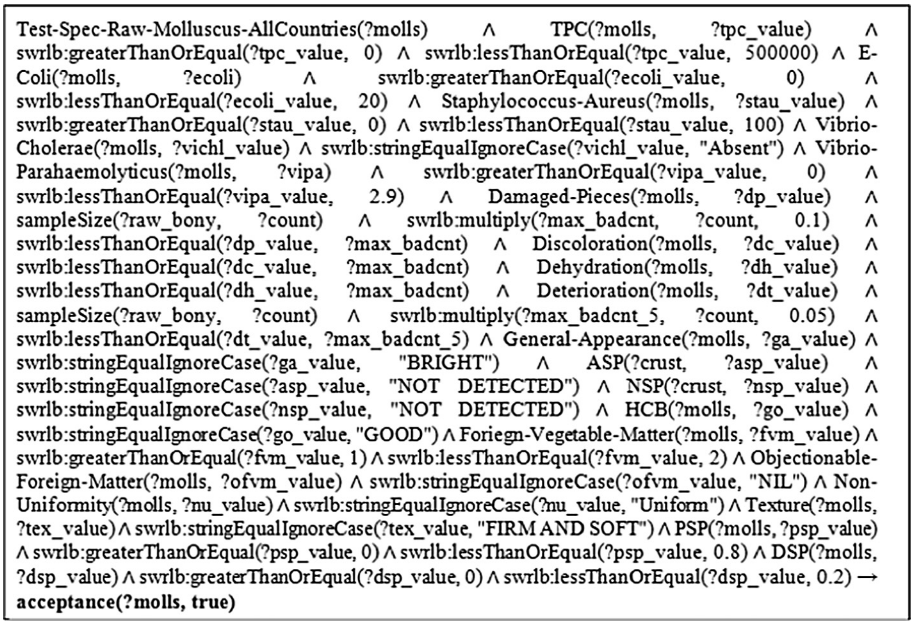
Table 4 presents the statistics of the data sets.

3.2. Performance metrics

The performance of the proposed comparison algorithm is evaluated using the statistical measures such as Precision, Recall, and Accuracy [5].

Precision is the ratio of the number of valid samples correctly retrieved to the total number of samples extracted as valid.

$$Precision = \frac{True\ Positive}{True\ Positive + False\ Positive}$$



```
Test-Spec-Raw-Molluscus-AllCountries(?molls) ∧ TPC(?molls, ?tpc_value) ∧
swrlb:greaterThanOrEqual(?tpc_value, 0) ∧ swrlb:lessThanOrEqual(?tpc_value, 500000) ∧ E-
Coli(?molls, ?ecoli) ∧ swrlb:greaterThanOrEqual(?ecoli_value, 0) ∧
swrlb:lessThanOrEqual(?ecoli_value, 20) ∧ Staphylococcus-Aureus(?molls, ?stau_value) ∧
swrlb:greaterThanOrEqual(?stau_value, 0) ∧ swrlb:lessThanOrEqual(?stau_value, 100) ∧ Vibrio-
Cholerae(?molls, ?vichl_value) ∧ swrlb:stringEqualIgnoreCase(?vichl_value, "Absent") ∧ Vibrio-
Parahaemolyticus(?molls, ?vipa) ∧ swrlb:greaterThanOrEqual(?vipa_value, 0) ∧
swrlb:lessThanOrEqual(?vipa_value, 2.9) ∧ Damaged-Pieces(?molls, ?dp_value) ∧
sampleSize(?raw_bony, ?count) ∧ swrlb:multiply(?max_badcnt, ?count, 0.1) ∧
swrlb:lessThanOrEqual(?dp_value, ?max_badcnt) ∧ Discoloration(?molls, ?dc_value) ∧
swrlb:lessThanOrEqual(?dc_value, ?max_badcnt) ∧ Dehydration(?molls, ?dh_value) ∧
swrlb:lessThanOrEqual(?dh_value, ?max_badcnt) ∧ Deterioration(?molls, ?dt_value) ∧
sampleSize(?raw_bony, ?count) ∧ swrlb:multiply(?max_badcnt_5, ?count, 0.05) ∧
swrlb:lessThanOrEqual(?dt_value, ?max_badcnt_5) ∧ General-Appearance(?molls, ?ga_value) ∧
swrlb:stringEqualIgnoreCase(?ga_value, "BRIGHT") ∧ ASP(?crust, ?asp_value) ∧
swrlb:stringEqualIgnoreCase(?asp_value, "NOT DETECTED") ∧ NSP(?crust, ?nsp_value) ∧
swrlb:stringEqualIgnoreCase(?nsp_value, "NOT DETECTED") ∧ HCB(?molls, ?go_value) ∧
swrlb:stringEqualIgnoreCase(?go_value, "GOOD") ∧ Foriegn-Vegetable-Matter(?molls, ?fvm_value) ∧
swrlb:greaterThanOrEqual(?fvm_value, 1) ∧ swrlb:lessThanOrEqual(?fvm_value, 2) ∧ Objectionable-
Foreign-Matter(?molls, ?ofvm_value) ∧ swrlb:stringEqualIgnoreCase(?ofvm_value, "NIL") ∧ Non-
Uniformity(?molls, ?nu_value) ∧ swrlb:stringEqualIgnoreCase(?nu_value, "Uniform") ∧ Texture(?molls,
?tex_value) ∧ swrlb:stringEqualIgnoreCase(?tex_value, "FIRM AND SOFT") ∧ PSP(?molls, ?psp_value)
∧ swrlb:greaterThanOrEqual(?psp_value, 0) ∧ swrlb:lessThanOrEqual(?psp_value, 0.8) ∧ DSP(?molls,
?dsp_value) ∧ swrlb:greaterThanOrEqual(?dsp_value, 0) ∧ swrlb:lessThanOrEqual(?dsp_value, 0.2) →
acceptance(?molls, true)
```

Figure 5 Test specifications – raw material – molluscs in SWRL.

Table 2 Data store update algorithm.

```
algorithm: dataStoreUpdate(test)
foreach test in testSet Array do
test_values[] $\leftarrow$ retreiveFromOntologyValues(test)
test_values [] $\leftarrow$ sort(test_values)
test_values [] $\leftarrow$ removeDuplicates(test_values)
//Set minimum value
if (min(test_values) $<$ acceptableMinValueFromOntology(test))
then
      min_value $\leftarrow$ min(test_values)
else
      min_value $\leftarrow$ acceptableMinValueFromOntology(test)
end if
//Set maximum value
if (max(test_values) $>$ acceptableMaxValueFromOntology(test))
then
      max_value $\leftarrow$ max(test_values)
else
      max_value $\leftarrow$ acceptableMaxValueFromOntology(test)
end if
avg_value $\leftarrow$ mean(test_values)        //Compute Variance
foreach item, $x_i$ in test_values do
temp $\leftarrow$ avg_value
temp $\leftarrow$ $(x_i - temp)^2$
mn_sq $\leftarrow$ (mn_sq + temp)
end for
variance $\leftarrow$ (mn_sq $\div$ count($x_i$))
writeToDataStore(min_value, max_value, variance, test)
next test
end
```

Table 3 Comparison algorithm.

```
algorithm: comparison(test, current_value)
min_value $\leftarrow$ getMinValueOfTest(test)
max_value $\leftarrow$ getMaxValueOfTest(test)
if ((current_value $\geqslant$ min_value) and (current_value $\leqslant$ max_value))
then
      output 'Known Value'
else
   range $\leftarrow$ (max_value $-$ min_value)    //Compute Range
   variance $\leftarrow$ getVariance(test)    // Compute Standard Deviation
std_dev $\leftarrow$ squareRoot(variance)
if (current_value $<$ min_value) then
   if (current_value $<$ (min_value $-$ std_dev)) then
      output "AcceptableLevel"
   else if (current_value $<$ (min_value $-$ range)) then
      output "Probable Error"
   else
      output "Unacceptable value! Check the input OR repeat the
test"
   end if
else if (current_value $>$ max_value) then
   if (current_value $<$ (max_value $+$ std_dev)) then
      output "Acceptable Level"
   else if (current_value $<$ (max_value $+$ range)) then
      output "Probable Error"
   else
      output "Unacceptable value! Check the input OR repeat the
test"
   end if
else
output "ERROR!!!"
end if
end if
end
```

Recall is the ratio of the number of valid samples correctly retrieved to the total number of valid samples in the dataset.

$$Recall = \frac{True\ Positive}{True\ Positive + False\ Negative}$$

Accuracy is a statistical measure to determine how well a test correctly identifies or excludes a condition. It is the proportion of true results among the total number of cases examined.

$$Accuracy = \frac{True\ Positive + True\ Negative}{True\ Positive + False\ Negative + False\ Positive + True\ Negative}$$

The tables below present the results. It shows the observed results and the predicted ones along with the performance metrics (see Tables 5–7).

For Bony Fish dataset, the algorithm identified 37 test samples correctly as true positives and ten samples as true negatives. It wrongly lists two valid test samples as having problems in the test values. The algorithm incorrectly outputs three invalid test samples as within the allowable limit.

For Cephalopods data set, the algorithm correctly listed 74 test samples and 12 test samples as true positives and true negatives respectively. It wrongly classified two positive test samples as negative, and five negative test samples as positive.

For Molluscs data set, the algorithm correctly outputs 15 test samples as known values. The algorithm correctly recognized 3 test samples as 'probable error' or 'unacceptable value'. One sample whose test value was in the acceptable range was output wrongly as 'probable error'. The number of samples whose test values were outside the range identified as 'known value' is 2.

Table 4 Comparison algorithm dataset.

Dataset	Number of test samples
Bony fish	52
Cephalopods	93
Molluscs	21

Table 5 Result of comparison algorithm on bony dataset.

		Predicted	
		Positive	Negative
Observed	Positive	37	2
	Negative	3	10
Precision		0.93	
Recall		0.95	
Accuracy		0.90	

3.3. Performance analysis and comparison

Fig. 6 shows the overall measures of Precision, Recall, and Accuracy in identifying the initial errors in test values using the proposed algorithms. The precision values of all data sets are obtained in the range 0.88–0.94 with the least precision

Table 6 Result of comparison algorithm on cephalopods dataset.

		Predicted	
		Positive	Negative
Observed	Positive	74	2
	Negative	5	12
Precision		0.94	
Recall		0.97	
Accuracy		0.92	

Table 7 Result of comparison algorithm on molluscs dataset.

		Predicted	
		Positive	Negative
Observed	Positive	15	1
	Negative	2	3
Precision		0.88	
Recall		0.94	
Accuracy		0.86	

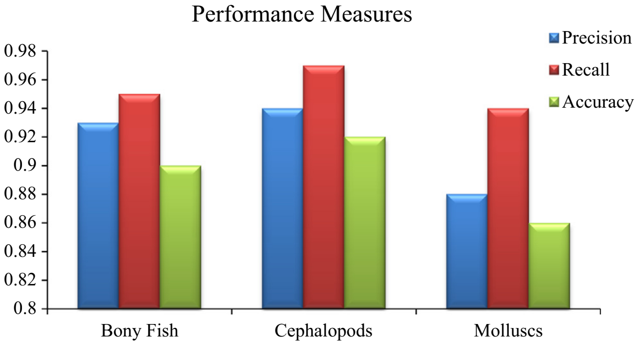


Figure 6 Performance measures – comparison algorithm.

value in the case of Molluscs data set. Recall measures are the highest (0.94–0.97) among the three measures for all data sets. Cephalopods data set resulted in the highest accuracy among the other two.

4. Conclusion

The paper presents the initial phase of a proposed model ONTO SQAM, an ontology-based model to ensure the quality of seafood. Two algorithms are proposed to check the parameter values at the initial stage. It helps to understand about any errors or quality issues at the beginning. The Data Store Update algorithm updates the acceptable minimum and maximum value of a test in the data store. The Comparison algorithm compares the new test values with these updated values to ensure that the test values are in the acceptable range. Later, the actual comparison of the test values with the test standards stored in the ontology is performed. The performance metrics such as precision, recall, and accuracy are used to evaluate the performance of the model. The experimental results obtained on the datasets show that all the three measures achieved good results.

Funding

This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

References

[1] Ying Ding, Schubert Foo, Ontology research and development. Part 2-a review of ontology mapping and evolving, J. Inf. Sci. 28 (5) (2002) 375–388.

[2] D. Fensel, Ontologies: A Silver Bullet for Knowledge Management and Electronic-commerce, Spring-Verlag, Berlin, 2000.

[3] M.D. Makwana et al., Review: Hazard Analysis and Critical Control Points in Meat Industry, 2015.

[4] Natalya F. Noy, Deborah L. McGuinness, Ontology development 101: a guide to creating your first ontology, 2001.

[5] Sebastian Raschka, An Overview of General Performance Metrics of Binary Classifier Systems, 2014. arXiv preprint arXiv:1410.5330.

[6] P.V. Vinu, P.C. Sherimon, K. Reshmy, Development of seafood ontology for semantically enhanced information retrieval, Int. J. Comput. Eng. Technol., IJCET (2012), ISSN: 0976-6367.

[7] P.V. Vinu, P.C. Sherimon, K. Reshmy, Modeling of test specifications of raw materials in seafood ontology using semantic web rule language (SWRL), in: Proceedings of the 2015 International Conference on Advanced Research in Computer Science Engineering & Technology (ICARCSET 2015), ACM, 2015.