

LOANONT-A RULE BASED ONTOLOGY FOR PERSONAL LOAN ELIGIBILITY EVALUATION

Neha Jain and Lalit Sen Sharma

Department of Computer Science and IT, University of Jammu, Jammu, India

ABSTRACT

In recent years, significant attention has been given to understand and implement banking solutions. The global competitive business environment and advancement in Information Technology and in particular internet technologies has facilitated the carrying out of banking activities outside the brick and mortar premise of the banks. Credit availing schemes are the core of the banking industry. Many agencies are working on it so as to make this facility hassle free for the customers and also to minimize the losses incurred by the banks in the form of bad debts. The challenge has been, and still is, to recognize, communicate and steadily improvise the banking solutions. The internet technologies are a potential candidates to overcome these challenges. The paper describes LoanOnt Ontology with the associated implementation toolset for creating an interoperable and sustainable personal loan calculation solution which would provide an intercommunication platform to facilitate integration and interoperation of information across interacting applications in banking scenarios.

KEYWORDS

Protégé, OWL (Web Ontology Language), SWRL(Semantic Web Rule Language), SQWRL(Semantic Query Enhanced-Web Rule Language).

1. INTRODUCTION

Sustainable and robust banking solutions are core of a healthy economy. In today's era of Information Technology and growing popularity of the e-commerce and e-business, the banking solutions also need to be made interoperable.

Ontologies are a promising solution to bind complexity, organize information, bringing web like qualities to the interlined data sets and provide an interoperable solution on the scale of the World Wide Web. This paper attempts to put a sample credit score calculation facility in the ontology framework. The Web Ontology Language is used for creating the ontology and Semantic Web Rule Language is used for enhancing its functionality with rules.

The idea developed in the form of LoanOnt, can be used by various credit rating agencies in collaboration with various credit lending institutions so that a unique score can be generated for an individual which can amalgamate various parameters that constitute the scoring model of a

credit lending institution and also the repayment of loans along with other parameters that constitute the criterion for calculating the credit score by a credit rating agency.

The layout of the paper is as follows: The ontology concept, the OWL language, SWRL language will be discussed in the second section followed by the LoanOnt Ontology and a discussion on its application areas.

2. TOOLS AND TECHNOLOGIES

An ontology is a formal, explicit specification of a shared conceptualization[8]. The concept 'Ontology' has its roots in Philosophical regime, in computer science arena, ontologies have surfaced from artificial intelligence. Ontologies are normally related with logical inferencing to offer a common perceptive of a particular domain. They facilitate the domain to be connecting people, organization and application systems. In nutshell, ontologies communicated are a promising and robust tool to provide assistance to human-machine communication, to realize interoperability between software systems and to make the software systems more robust and resilient.

Many formal languages have been developed to construct ontologies. These languages facilitate knowledge encoding concerning explicit domains and frequently include reasoning capabilities that provide for the processing of the encoded knowledge. These languages are generally declarative, built on first order logic or description logic-[6]. The language chosen for developing LoanOnt is W3C Web Ontology Language (OWL). It is a Semantic Web language intended to render a loaded and complex knowledge concerning things, their groups, and relations between them. OWL is a computational language which is rooted in logic. Knowledge manifested in OWL can be utilized by computer programs, e.g., to prove the knowledge consistency or to frame the implicit knowledge explicit. The documents of OWL, preferably called ontologies, can be published in the World Wide Web and may refer to or be referred from other OWL ontologies. OWL is a component of the Semantic Web technology store of the World Wide Web Consortium, which comprises RDF, RDFS, SPARQL, etc.[1]. The syntax for saving the OWL ontology is chosen to be RDF/XML as this syntax is compliant with all conformant OWL2 softwares[2].

To add rules to the ontology, Semantic Web Rule Language (SWRL) is used. SWRL is a non-procedural high level language belonging to the category of the fourth generation languages. It has as its core an integration of the OWL DL and OWL Lite sublanguages of the Web Ontology Language with the Unary/Binary Datalog RuleML sublanguages of the Rule Markup Language. SWRL expands the set of OWL axioms to encompass Horn-like rules. The framework of the suggested rules is an implication between an antecedent (body) and consequent (head), both of which are composed of zero or more atoms. The intentional sense conveyed is as: whenever the conditions specified in the antecedent are held true, then the conditions specified in the consequent must also be true[4]. An empty antecedent is obligated as being trivially true (i.e. satisfied by every interpretation), so the consequent must also be satisfied by every interpretation; an empty consequent is treated as trivially false (i.e., not satisfied by any interpretation), so the antecedent must also not be satisfied by any interpretation.

To query the SWRL based OWL ontologies, the SQWRL query language is used. Semantic Query-enhanced Web Rule Language, acronym as SQWRL, has its base in the SWRL language. SQWRL takes a standard SWRL rule antecedent and effectively treats it as a pattern specification for a query. It replaced the rule consequent with a retrieval specification. SQWRL's syntax is also very similar to that of SWRL. Standard SWRL serialization mechanisms can be implemented to store the SQWRL queries in OWL ontologies.[5]

Capturing ontology involves an in-depth study of the domain of interest and subsequently, identification and classification of the key terms, concepts and associations between the terms. The production of unambiguous text definitions of the identified concepts is also an important step in ontology capturing. These modeling errands are carried out using Protégé. It provides a user friendly graphical user interface for creating knowledge-based applications and domain knowledge model. It maintains a set of tools to aid the carrying out of various tasks like Ontology visualization, reasoning, ontology export, ontology import, inferencing, ontology querying etc. [7]. The individual information is fed to the ontology using the Cellfie plugin which is a tool for creating ontologies from spread-sheets.

3. LOANONT OVERVIEW

LoanOnt is the outcome of the research work undertaken. It enables a common formal representation of information in Personal loan systems, hence creating a semantic base for application integration. It is a novel effort towards making the banking services interoperable, hassle free for the customers, and to minimize the losses incurred by the banks in the form of bad loans. LoanOnt is a rule based ontology developed using Protégé in OWL language. Individual information to determine the eligibility for loan procurement is added to the ontology using the Cellfie plugin of Protégé. The ontology calculates an individual's score based on parameters like age, Academic qualification, association with bank, nature of employment, asset value, liability count etc. The parameters are carefully chosen to assert the repayment capacity of the customer so as to evade the risk of bad loans. These parameters are set as data type properties.

The rules are framed and the ontology is enriched with them using the SWRL tab of the protégé ontology development tool.

The rules are of the form

Person(?p) \wedge Age(?p, ?age) \wedge swrlb:lessThan(?age, 30) -> AgeScore(?p, 8)

A set of rules for different age groups has been formulated.

Rules for Academic qualification have been formulated as:

Person(?p) \wedge AcademicQualification(?p, "Doctorate") -> AcademicQualificationScore(?p, 8)

Rules for the association with the bank have been constructed as:

Person(?p) \wedge AssociationWithBank(?p, true)^NoOfYearsOfBankAssociation(?p,?num) \wedge swrlb:lessThanorEqual(?num,5) -> BankAssociationScore(?p, 5)

Similarly, rules have been formulated for all the criteria, considering which, the eligibility score is ascertained. Semantic Web Rule Language is used for writing the rules for the ontology. The ontology creates ABox and TBox assertions [1]. The TBox contains the described domain structure of the ontology and the ABox contains the real objects. The TBox contains the intentional knowledge, which is general knowledge concerning the domain of discourse. In the LoanOnt, the TBox contains the properties and the rules specified.

4. METHODOLOGY

The LoanOnt ontology model described in this research study is inspired by the methodology proposed by Ushold and King [9]. This methodology consists of broadly four stages: identify the purpose of the ontology; build the ontology; evaluate the ontology; and finally document it. The building stage is further divided into three major stages, namely the requirements capturing stage, coding stage and integrating with other ontologies.

4.1 Identifying the purpose of LoanOnt

The impetus behind creating LoanOnt is to provide an interoperable solution to the personal loan eligibility determination process. In today's era of Information Technology and Web Technologies, the banking solutions also need to be made interoperable. A lot many credit agencies provide customers with their credit history and credit score. This score is based on the repayment of various credit facilities availed by the customers. A customer can check his credit score and take corrective actions where required. The credit score also provides a reference to the lending institutions while sanctioning credit facilities. Various lending institutions have also devised their own credit scoring models which they use while approving credit facilities to customers. The credit score adds objectivity to otherwise subjective underwriting process.

4.2 Building LoanOnt

Ontology building stage includes three major tasks:

4.2.1 Capturing LoanOnt

Capturing an ontology encompasses the identification and classification of concepts and terms pertaining to the key ideas and the associated relationships between them. This development stage includes the creation of the related unambiguous and concise text definitions. Protégé offers a set of tools to aid the creation, visualization and manipulation of ontologies in various representation formats. The top-level class in LoanOnt is the Person class. The Person class has many data properties like AcademicQualification, Age, AnnualIncome, associationWithBank, DependentCount, NetWorth etc.

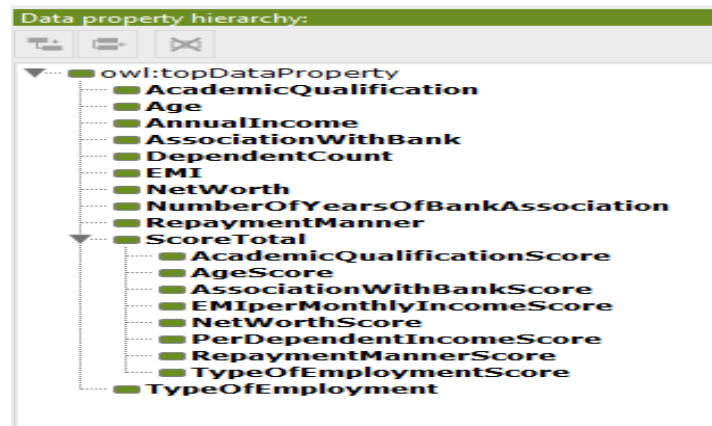


Figure 1: Basic data properties of Person class

The AcademicQualification data property refers to the academic qualification of the applicant. The Academic QualificationScore is ascertained based on this property value using SWRL rules. Similarly, other properties are used to calculate their respective scores and finally the scores are added up, again using SWRL rules, to assign value to the ScoreTotal data property.

4.2.2. Coding LoanOnt

Ontology coding refers to explicitly and clearly representing the conceptualization captured in the prior stage in some formal language. Broadly categorizing, the ontology representation languages can be enlisted under two distinguished heads[10]. The first list is formed of several ontology languages that are obtained from research efforts of knowledge description paradigm in the Artificial Intelligence community and are not intended for use in the web, such as KIF, Ontolingua, OCML, FLogic and Loom. The second list is generally known as the web-based ontology language, comprising OIL, DAML, RDFS, DAML+OIL, SHOE and OWL. The first five enlisted ontology languages are all precursors of OWL and form the base on which the state of the art formal language has been built upon [11]. In this work, OWL is used to formalize the LoanOnt considering the entailing particulars. Primarily, OWL is an ontology language which is mainly oriented for use for web, that uses to good advantage the disparity of the web and hence when contrasted with the first group, it is extra robust and more promising for web based systems to provide an interoperable web based system. Secondly, OWL is a very refined formal language that shares numerous design features and useful language constructs with its precursors, retaining utmost compatibility with them. It also adds attractive characteristics through suitable additions to meet a large number of sometimes disagreeing requirements like accord between expressive power and inference support. Lastly, OWL embraces a description logic (DL)[12] style model theory to illustrate and formalize the meaning of the language, which assists ontologies, and information taking vocabulary represented by ontologies, to be exchanged and distributed with consistent and precise semantics. The DL style model also allows automated reasoning and such reasoning techniques can be employed to figure out the consistency of the ontology and also to calculate the classification hierarchy [11].

4.2.3. Coding LoanOnt in OWL DL

The key elements in LoanOnt are outlined by class axioms and property axioms in OWL DL. For example, the *Person* class is explicitly defined in OWL DL RDF/XML syntax as follows:

```
<owl:Class
rdf:about="http://www.semanticweb.org/dell
/ontologies/2016/5/untitled-ontology-107
#Person">
  <rdfs:subClassOf
rdf:resource="http://www.w3.org/2002/07/ow
l#Thing"/>
</owl:Class>
```

Here, *Thing* is a predefined, built-in class in OWL, regarded as the super-class of all OWL classes. These RDF/XML exchange syntax definitions can be communicated in OWL DL abstract syntax, expressing the same semantics.

LoanOnt defines various properties for the Person class. These properties have varied values. An analysis of these values is done and a score is assigned to each individual depending upon the property value. The score assigned to all property values is summed up to generate the final score, the decision for personal loan eligibility is based upon this final score.

At this stage, a limitation of OWL was encountered. The essential requirement of the proposed ontology is that it requires consideration of more than one property at a time. Although OWL offers a rich capability for class constructors, expressiveness with regards to properties is inadequate. Although OWL allows chaining of properties, it does not support making of assertions in relation to the equality of the objects at the closing stages of two different properties or property chains[13]. Since, precisely this kind of assertions are a mandatory requirement of the system under consideration, it has to be assisted with some add on to OWL.

The Semantic Web Rule Language (SWRL) is used to deal with the shortcomings of OWL. SWRL has its basis in the OWL DL and OWL Lite sublanguages of the OWL Web Ontology Language with the Unary/Binary Datalog RuleML sublanguages of the RuleML sublanguages of the Rule Markup Language. SWRL covers a high-level abstract syntax for Horn-like rules in the two of the OWL DL and OWL Lite sublanguages of OWL [13].

For example: Considering the age parameter, AgeScore is assigned according to the age of the applicant. The rules are of the form:

```
Person(?p)  $\wedge$  Age(?p, ?age)  $\wedge$  swrlb:lessThanOrEqual(?age, 30)  $\wedge$  swrlb:greaterThan(?age, 25)
-> AgeScore(?p, 9)
```

Built in functions of SWRL are prefixed with 'swrlb' to use the functionality.

Another parameter chosen is Academic Qualification. AcademicQualificationScore is assigned depending upon the value of this parameter. The rules are of the form:

Person(?p) ∧ AcademicQualification(?p, ?edu) ∧ swrlb:containsIgnoreCase(?edu, "doctorate") -> AcademicQualificationScore(?p, 9)

Next determining parameter is Association With Bank. AssociationWithBank Score is calculated depending upon the number of years of association with the bank:

Person(?p) ∧ AssociationWithBank(?p, true) ∧ NumberOfYearsOfBankAssociation(?p, ?n) ∧ swrlb:lessThanOrEqual(?n, 5) -> AssociationWithBankScore(?p, 10)

Another decisive factor is Type of Employment. TypeOfEmployment Score is based upon the value of this factor.

Person(?p) ∧ TypeOfEmployment(?p, ?edu) ∧ swrlb:containsIgnoreCase(?edu, "PSU") -> TypeOfEmploymentScore(?p, 14)

Next vital parameter is Net Worth. NetWorthScore is calculated based upon this parameter.

Person(?p) ∧ NetWorth(?p, ?wrth) ∧ swrlb:lessThan(?wrth, 300000) ∧ swrlb:greaterThanOrEqual(?wrth, 100000) -> NetWorthScore(?p, 4)

Next factor is Income Per Dependent. PerDependentIncome Score is calculated using rules of the form:

Person(?p) ∧ DependentCount(?p, ?dep) ∧ AnnualIncome(?p, ?in) ∧ swrlb:divide(?depInc, ?in, ?dep) ∧ swrlb:lessThan(?depInc, 25000) -> PerDependentIncomeScore(?p, 0)

Another key factor chosen is the mode of repayment of loan. RepaymentMannerScore is calculated based upon this factor using rules of the form:

Person(?p) ∧ RepaymentManner(?p, ?repay) ∧ swrlb:containsIgnoreCase(?repay, "cash") -> RepaymentMannerScore(?p, 0)

The EMI chosen and its ratio to the monthly income is also an important factor for determining the repayment capacity and to defer bad loans. EMIPerMonthlyIncome score is calculated considering this parameter. The rules are of the form:

Person(?p) ∧ EMI(?p, ?emi) ∧ AnnualIncome(?p, ?ann) ∧ swrlb:divide(?inc_monthly, ?ann, 12) ∧ swrlb:divide(?emiPerMonth, ?emi, ?inc_monthly) ∧ swrlb:greaterThan(?emiPerMonth, 0) ∧ swrlb:lessThanOrEqual(?emiPerMonth, 0.1) -> EMIPerMonthlyIncomeScore(?p, 16)

The above mentioned form of rule set is determined and written in SWRL. The Pellett reasoner is used to do the consistency check. Individual parameter scores are added together using SWRL built in function and final score is calculated which is used to determine the loan eligibility.

The LoanOnt model is expressed using the Protégé-OWL plug-in. An OWL document is created to elicit all the formal definitions. The document is saved in RDF/XML format and can be integrated with any Java application by using OWL-AP which is a Java API. It is a reference implementation for facilitating the creation, manipulation and serialization of OWL ontologies. It also provides an API for OWL 2. It is open source and is available under Apache Licenses. It also contains an efficient in-memory reference implementation, an RDF/XML parser and writer, an OWL/XML parser and writer, OWL-Functional Syntax parser and writer, Turtle parser and writer, KRSS parser and writer and OBO Flat file format parser. Reasoner Interfaces are also integrated with OWL API for incorporating reasoners such as FaCT++, HermiT, Pellet and Racer.

Another contemporary Java programming toolkit available is *Jena*. It offers a consistent programming interface to facilitate ontology application development, independent of the underlying ontology language. A range of reasoners is also supported by Jena via the interface API [13].

4.2.4. Integrating other ontologies

LoanOnt represents general information in Personal Loan eligibility checker systems, hence providing a syntactic and semantic groundwork for application integration. LoanOnt defines basic concepts and relations common to personal loan systems, which can be instantiated or enhanced to generate new relations to describe knowledge related to application of particular loan systems. A Protégé Desktop plugin for mapping spreadsheets to OWL ontologies called Cellfie is used to create ontologies from spreadsheets. A very flexible mapping mechanism is provided by the Cellfie plugin for the creation of OWL ontologies from spreadsheets. The LoanOnt ontology serves as our initial ontology to provide the frame about the customer information. The creation of the Transformation Rules using the *MappingMaster DSL* syntax and functions is the key task of this ontology creation task.

The transformation rules are created in an editor provided with the Cellfie plugin. The rules are saved as JSON files. Axioms are generated based on the transformation rules.

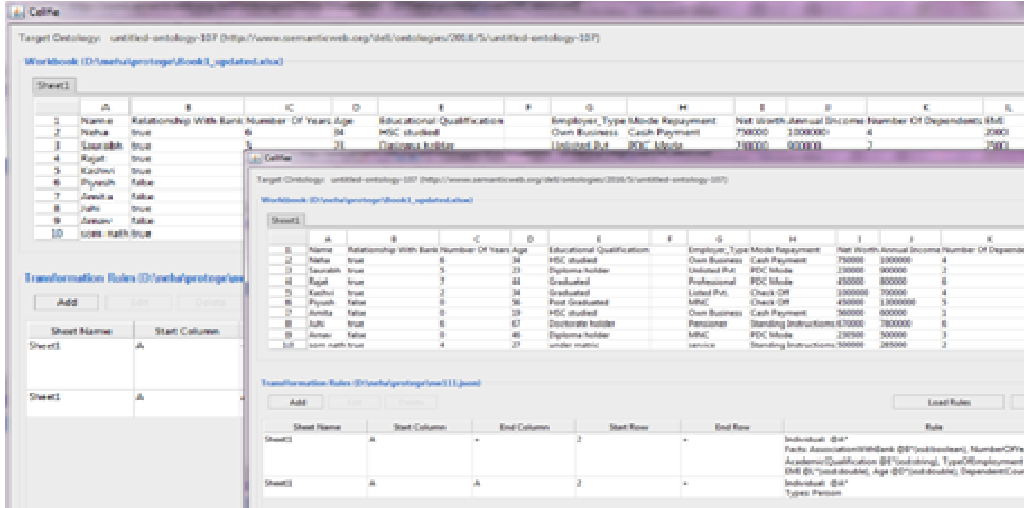


Figure 2: Using Cellfie

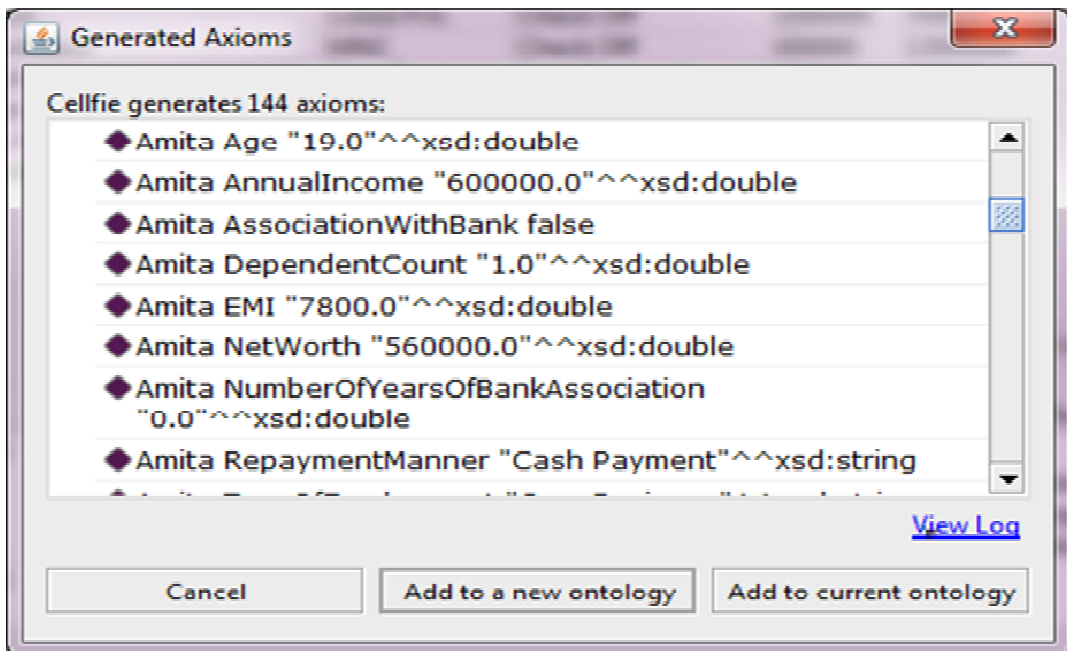


Figure 3: The axioms generated using the Cellfie plugin

The generated axioms can either be inserted to the current active ontology or a new ontology can be created and all the generated axioms can be inserted there. The active ontology becomes an imported ontology which is automatically embedded into the new ontology.

5. EVALUATING THE ONTOLOGY:

The Ontology is integrated with a JAVA application using the OWL API. It is queried using SQWRL queries. Different scores based upon varied parameters are calculated both manually and also using the LoanOnt. The consistency of the two results determine the accuracy of the LoanOnt.

For example, considering the case of a customer who is 34 years in age, has a bank account since last two years, has graduated and is working in a listed private company, agrees to pay rs.1000 per month as EMI as check off, has an annual income of 200000 has a net worth of 1000000, and has four dependents. Calculating manually, her score comes out to be 77 which is in consistency with the score calculated using LoanOnt.

Using LoanOnt, following type of SQWRL query is used to generate the final total score:

```
Person(?p) ^ AgeScore(?p, ?age) ^ AcademicQualificationScore(?p, ?academicQual) ^
AssociationWithBankScore(?p, ?bankAssociation) ^ EMIPerMonthlyIncomeScore(?p, ?emiPerIncome) ^
NetWorthScore(?p, ?netWorth) ^ PerDependentIncomeScore(?p, ?perDepIncome) ^
RepaymentMannerScore(?p, ?repaymentMode) ^ TypeOfEmploymentScore(?p, ?employmentType) ^
swrlb:add(?scoreTotal, ?age, ?academicQual, ?bankAssociation, ?emiPerIncome, ?netWorth,
?perDepIncome, ?repaymentMode, ?employmentType) -> sqwrl:select(?p, ?age, ?academicQual,
?bankAssociation, ?emiPerIncome, ?netWorth, ?perDepIncome, ?repaymentMode, ?employmentType,
?scoreTotal)
```

p	age	academicQual	bankAssociation	emiPerIncome	netWorth	perDepIncome	repaymentMode	employmentType	scoreTotal
Neha	10	2	15	10	4	4	0	0	45
Amev	5	5	0	5	4	0	15	14	48
somNath	9	2	10	16	2	15	5	0	59
Piyush	3	9	0	16	2	15	9	14	68
Juhi	1	9	15	12	2	15	5	20	79
sanjeev	9	0	10	16	7	15	15	0	72
Saurabh	8	5	10	16	4	15	15	8	81
anikush	10	5	15	16	4	15	15	0	80
Amita	8	2	0	12	2	15	0	0	39
Kashvi	10	5	10	16	7	6	9	14	77
Rajat	7	5	15	12	2	6	15	8	70

Figure 4: Final Score calculated by LoanOnt

6. CONCLUSION

The research presents a practical implementation of one of the ontology development methodologies. The methodology proposed by Ushold and King has been used for developing semantic ontology models in e-banking domain. LoanOnt might demonstrate the interoperability of the e-banking services and might further seed the development of ontologies in the chosen domain, encouraging the use of the existing ontology development methodologies in the Semantic Web development processes of e-banking domain ontologies and further support the implementation of semantic technologies in e-banking domain.

Although, LoanOnt has been developed in e-banking domain, it also signifies an input to the ontology engineering field and also to Semantic Web domain. The framework and techniques employed provide a practical implementation of the Ushold and King methodology. The scope of the framework is not limited to the chosen domain but can be easily extended to other knowledge domains.

REFERENCES:

- [1] World Wide Web consortium Web Ontology Language Reference manual <https://www.w3.org/2001/sw/wiki/OWL>
- [2] W3C OWL2 Reference Document https://www.w3.org/TR/2012/REC-owl2-overview-20121211/#Documentation_Roadmap
- [3] Wikipedia Declarative Programming page https://en.wikipedia.org/wiki/Declarative_programming
- [4] W3C SWRL document <https://www.w3.org/Submission/SWRL/>
- [5] Connor O' Martin & Das Amar,(2009)"SQWRL: a Query Language for OWL", OWL: Experiences and Directions (OWLED), Fifth International Workshop
- [6] Wikipedia Ontology Language document: https://en.wikipedia.org/wiki/Ontology_language
- [7] Protégé: <http://protege.stanford.edu/>
- [8] T. R. Gruber,(1993),"A Translation Approach to Portable Ontologies", Knowledge Acquisition, 5(2):199–220.
- [9] Ushold Mike&King Martin(1995),"Towards a Methodology for Building Ontologies", Workshop on Basic Ontological Issues in Knowledge Sharing, held in conjunction with IJCAI-95
- [10] Corcho, O., Fernandez-Lopez, M. & Gomez-Perez, A. (2003),"Methodologies, Tools and Languages for Building Ontologies: Where is their meeting point?", Data & Knowledge Engineering, 46: 41–64.
- [11] Baader, F., Calvanese, D., McGuinness, D.L., Nardi, D. and PatelSchneider, P.F.(2003), The Description Logic Handbook: Theory, Implementation, and Applications, Cambridge University Press: Cambridge.
- [12] OWL API documentation : <http://owlapi.sourceforge.net/>
- [13] Jena Ontology API: <https://jena.apache.org/documentation/ontology/>

AUTHORS

1. Neha Jain
Junior Research Fellow, Department of CS and IT, University of Jammu, Jammu.
2. Prof. Lalit Sen Sharma.
Professor, Department of CS and IT, University of Jammu, Jammu.