A Study on Ontology Life Cycle & Co-Relation Matrix

[1]Manjunath A, [2]Manjunath T N, [3]Chandrashekar K T
[1]Dept of ISE, BMSIT&M, Bangalore
Karnataka, India
manjutheboss07@gmail.com

[2]Dept of ISE, BMSIT&M, Bangalore
Karnataka, India
manju.tn@gmail.com

[3]Dept of ISE, BMSIT&M, Bangalore
Karnataka, India
chandru@bmsit.in

ABSTRACT: In modern virtual world, knowing each other and understanding is very difficult. In this paper we present a study on ontology life cycle which would help us in understanding the human emotions. Ontology is an effective computing paradigm which made it possible to include findings from social and psychology. The new method for emotion detection from text which depends on ontology extraction from the input sentence by using a triplet extraction algorithm. Then make an ontology matching with the ontology base that we created by similarity and word sense disambiguation. This ontology base consists of ontologies and the emotion label related to each one. We choose the emotion label of the sentence with the highest score of matching. If the extracted ontology doesn't match any ontology from the ontology base we use the keyword-based approach. This method doesn't depend only on keywords like the previous approaches it depends on the meaning of sentence words and the syntax and semantic analysis of the context. Ontology-based applications play an increasingly important role in the public and corporate Semantic Web. Methodologies used for the development of knowledge-based applications focus purely on knowledge engineering. Architectures for semantic web services involve ontologies, but naturally focus on services. For instance, WSMO or ODE-SWS provide ontology-based mechanisms to formally describe services and arriving at co-relation matrix between emotion classes.

Keywords: Ontology, Co-relation Matrix, Behavioral System

1. INTRODUCTION

Ontology engineering life cycle vary, they agree on the main lifecycle activities, namely

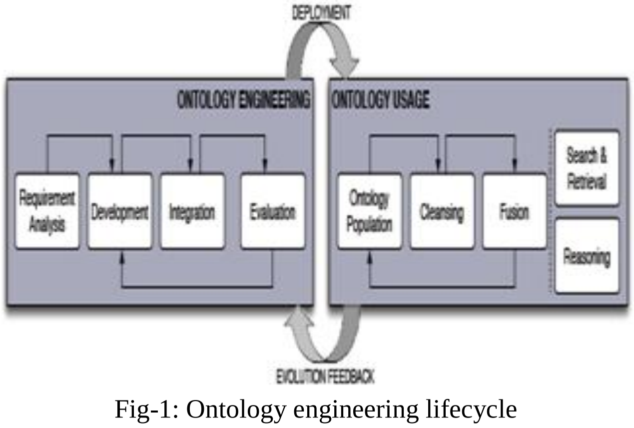

Fig-1: Ontology engineering lifecycle

As shown in fig-1, requirement analysis, development, evaluation, and maintenance, plus orthogonal activities such as project management. In the following:

Requirement Analysis: In this step, domain experts and ontology engineers analyze scenarios, use cases, and, in particular, intended retrieval and reasoning tasks performed on the ontology.

Development: This is the step in which the methodologies vary most. We therefore present an aggregated view on the different proposals for ontology development. The initial step is the identification of already available reusable ontologies and other sources such as taxonomies or database schemas. Once reusable ontologies are found, they have to be adapted to the specific requirements of the application this may include both backward and forward engineering of these reusable ontologies with respect to some design patterns. Then, the ontologies are translated to the target representation language. Because of the expressivity-scalability tradeoff involved in reasoning, it may be desirable to tweak the degree of axiomatization, e.g. for performance. An important aspect in development is collaboration. Existing proposals for reaching consensus knowledge involve the assignment of roles and the definition of interaction protocols for knowledge engineers.

Integration: Inspired by the componentization of software, recent approaches advocate the modularization of ontologies. Accordingly, the result of the development step shall be a set of modularized ontologies rather than one monolithic ontology. These modules have to be integrated, e.g. via the definition of import declarations and alignment rules. This integration concerns not only the modules that have been developed for the given use case. For interoperability with external applications, they may be embedded in a larger context, e.g. integrated with ontologies employed by other OIS.

Evaluation: Similar to bugs in software, inconsistencies in ontologies impede their proper use. So the initial evaluation step is to check for inconsistencies, both at the level of modules and in an integrated setting. Furthermore, ontologies also have to be assessed with respect to specific requirements.

1.1 Ontology Usage

It encompasses all activities performed with ontology after it has been engineered with all possible data set. So far, the lifecycle as described in the fig-1 is more of a static nature, just like the software lifecycle. Namely, if all requirements are met, the ontology will be deployed and the lifecycle continues with ontology evolution also referred to as maintenance. In this phase, new requirements may arise which are fed back into the loop, e.g. incorporated into the next release, which is then redeployed. Current lifecycle models however do not incorporate activities involved in the actual usage of ontologies. Through the life cycle we can search and retrieval and reasoning: Once the ontologies have been created, they can be used to realize information access in the application, for example via search and retrieval.

An OIS - Object Interface System, involves a reasoned to infer implicit knowledge. The schema can be combined with instance data to support advanced retrieval. The word ontology can designate different computer science objects depending on the context. It is important to distinguish these different forms of ontologies to clarify their content, their use and their goal. It is also needed to define precisely the vocabulary derived from the word ontology.

1.2 Information Ontologies

Information ontologies are composed of diagrams and sketches used to clarify and organize the ideas of collaborators to evolve the ontology dictionary. These ontologies are only used by humans. The characteristics of information ontologies are:
• Easily modifiable and scalable
• Synthetic and schematic
• Design process of the model

For instance, information ontology can be used during the conception phase of information system development or during the design of floor plan in architectural construction project. As shown in Fig-.2, information ontologies focus on concepts, instances and their relationships. Their goal is to propose an overview of a current project in order to express the state of this project. The grey color of the property elements means that properties are not always well defined by information.

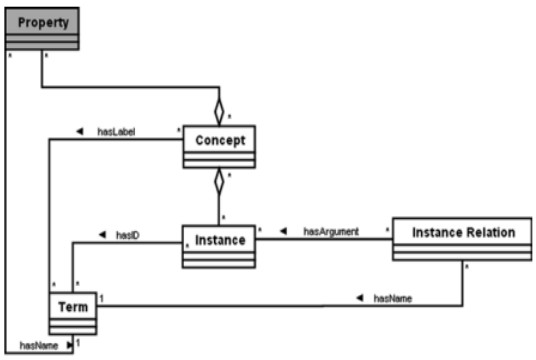

Fig-2: Ontology relations

1.3 Linguistic/Terminological Ontologies

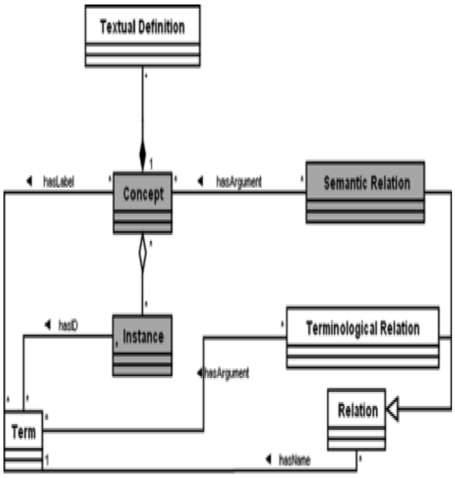

Fig-3: UML schema of software ontology components and their relationships

Linguistic ontologies can be glossaries, dictionaries, controlled vocabularies, taxonomies, folksonomies, thesauri, or lexical databases. As shown in Fig. 1.3 this type of ontology mainly focuses on terms and their relations Software Ontologies Software ontologies (or software implementation driven ontologies) provide conceptual schemata whose main focus is normally on data storage and data manipulation, and are used for software development activities, with the goal of guaranteeing data consistency. As shown in Fig-.3, a concept is composed of a set of properties; all concepts are also defined thanks to each other's by the relations they have. These relations are also associated to constraints. At execution time, data are stored in the properties of object, that is to say an instance of a concept. Thus, data could be processed in various treatments Nevertheless, software Fig-3 UML schema of software ontology components and their relationship.Rousse et al. ontologies goal is not to describe particular instances during execution time. Software ontologies are normally defined with conceptual modeling languages used in software and database engineering. These languages are used during software design procedure: for example Entity-Relationship Model language or Object Model Language. The next section presents the most well-known one called UML. UML presentation will be followed by one example of software ontology is used in building construction.

1.4 Unified Modeling Language (UML)

The Unified Modeling Language (UML) is a standard used mainly for modeling software and information systems. UML is a graphical language for visualizing, specifying and constructing any parts of software components. UML is a semi-formal formalism, because the official document defining the semantics of UML is mainly composed of informal descriptions in English.

Formal Ontologies require a clear semantics for the language used to define the concept, clear motivations for the adopted distinctions between concepts as well as strict rules about how to define concepts and relationships. This is obtained by using formal logic (usually first order logic or Description Logic) where the meaning of the concept is guaranteed by formal semantics (Borgo 2004).Ontology type is the only one that contains logical definition

1.5 Web Ontology Language (OWL)

The OWL Web Ontology Language is a standard recommended by the W3C. It is designed for use by applications that need to process the content of information instead of just presenting information to humans. OWL facilitates greater machine interpretability of Web content than that supported by XML, RDF, and RDF Schema (RDF-S) by providing additional vocabulary along with a formal semantics.

1.6 Depth in Ontology

Depth is also needed as it gives an idea about how specific is the emotion word with respect to its corresponding ontology structure and at what level that word falls. More specific smaller the depth, greater the weight age. Weight age value is calculated simultaneously while traversing the ontology tree. Emotion Ontology is an explicit specification of conceptualization. Ontologies have definitional aspects like high level schemas and aspects like entities and attributes interrelationship is between entities, domain vocabulary. Ontology allows a programmer to specify, in an open, meaningful way the concepts and relationships that collectively characterize some domain. Emotion can be expressed as joy, surprise, sadness, hate, fear, anger and so on. Since various emotion hierarchies are developed by researchers and there is not any standard emotion word hierarchy, focus is on the related research about emotion in cognitive psychology domain.

2. Co-Relation Matrix

Experiment-1

All the blogs are read and classified by manual assignment and their summary is shown in table-1. Total 135 blogs are taken and analyzed to test the Emotion Detection System. Table-1 shows number of blogs which fall in particular primary emotion class by manual assignment. In the experiment-2 the same blogs are used in emotion detector system as input to find the out in form of primary emotion classes.

TABLE-1.CLASSIFICATIONSUMMARY OBTAINED MANUALLY

Love	Joy	Sadness	Fear	Anger	Surprise
26	31	17	31	28	2

Experiment - 2

The emotion detector system has output file containing the emotion class of every blog as an output of the Emotion Detector System. Output of Emotion Detector System is shown in table-2 in the form of particular emotion class.

TABLE-2.CLASSIFICATION SUMMARY OF THE BLOGS BY USING THEPROPOSED FRAMEWORK

Love	Joy	Sadness	Anger	Fear	Surprise
30	27	21	25	30	2

After collecting number of blogs per emotion class from both experiments, we make a comparison bar chart shown in figure-1 between results obtained from Emotion Detection System and the results previously assigned by manual assignment. In the graph blue bar represents the number of blogs classified manually in primary emotion class and red bar represents the number of blogs classified by Emotion Detector System in primary emotion class. Horizontal axis represents the name of primary emotion class and vertical axis represents the number of blogs which fall in a particular primary emotion class. In the next section these results will be analyzed in form of precision and accuracy.

3. Results & Discussions

To test the detection capability and analyzing the results obtained from emotion detector system with ontology life cycle and dictionary, we need to check the accuracy of proposed framework. Binary classification test is the best way to check that how correct results are produced by Emotion Detector System. The decision matrix is used to calculate the classification accuracy, shown in table 3.

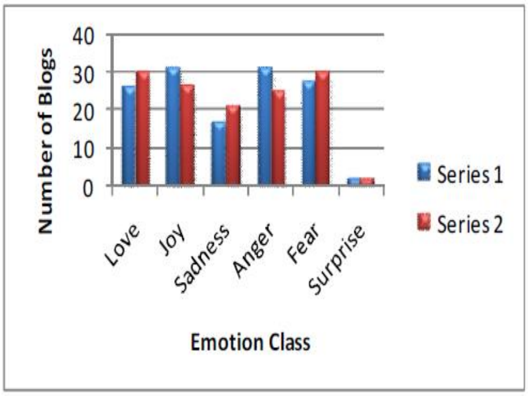

Fig-4: Comparison of the experiments

TABLE 3: CO-RELATION MATRIX FOR CALCULATING THE ACCURACYOF THE PROPOSED SYSTEM

	YES	**NO**
YES	**a**	**b**
NO	**c**	**d**

The meaning of a, b, c and d are defined below:
a: Proposed System and the manual assignment agree with the assigned emotion class,
b: Proposed System disagree with the assigned emotion class but the manual assignment agree,
c: Proposed System agree with assigned emotion class but the manual assignment disagree,
d: Proposed System and manual assignment both disagree.

4. CONCLUSIONS

Experiments have been done to test the proposed system in different condition to prove the effectiveness of the system and successfully completed all the tests with required results. Table 4 shows the accuracy and precision calculated for the emotion detector system. The test cases include running the Emotion Detector System and comparing the results generated in the form of emotion class with the emotion class of the blog assigned previously as titles. Finally number of blogs which emotion classes found same by Emotion Detector System as previously assigned as titles is called as "correct". Total 135 blogs are tested and 116 blogs' emotion classes are found correct by Emotion Detector System. These results proved that the proposed system and the implementation and testing of the Emotion Detector System are successful and it satisfies all the requirements. Text-based input is the most common way for humans to interact with computers while writing letters or giving feedback to any product in the era of web 2.0. Thus emotion detection from text focuses as an important research issue in affective computing. In this paper, existing research of emotion recognition based on textual data is surveyed and limitations of existing methods are reviewed. Emotion recognition system architecture is proposed to improve detection capabilities in an efficient manner. Proposed system is based on keyword spotting technique that is having rich features of ontology. Not all the limitations of existing methods are overcome by this architecture but use of ontology improves the detection capability by applying semantic approach.

4. REFERENCES

[1]. An Introduction to Ontologies and Ontology Engineering, Catherine Roussey, Francois Pinet, Myoung Ah Kang, and Oscar Corcho

[2]. METHONTOLOGY: From Ontological Art Towards Ontological Engineering, Mariano Ferndndez, Asunci6n G6mez-P~rez, Natalia Juristo

[3]. Towards an Ontology for Describing Emotions. Juan Miguel López, Rosa Gil, Roberto García, IdoiaCearreta, and Nestor Garay

[4]. Lifecycle-Support in Architectures for Ontology-Based Information Systems, Thanh Tran and Peter Haase and HolgerLewen and Óscar Muñoz-García and Asunción Gómez-Pérez and Rudi Studer Institute AIFB, Universität Karlsruhe, Germany

[5]. R. L. Breiger, *The Analysis of Social Networks*. London: Sage Publications Ltd, 2004, pages 505-526 in Handbook of Data Analysis,edited by Melissa Hardy and Alan Bryman.

[6]. T.Berners-Lee, "Semantic web road map," online at http://www.w3.org/DesignIssues/Semantic.html, 1998.

[7]. A. Swartz and J. Hendler, "The semantic web: A network of content for the digital city," in *Proceedings of Second Annual Digital CitiesWorkshop*, Kyoto, Japan, October 2001.

[8]. J. Hendler, T. Berners-Lee, and E. Miller, "Integrating applications on the semantic web," *Journal of the Institute of Electrical Engineers ofJapan*, vol. 122, no. 10, pp. 676–680, 2002.

[9]. L. Freeman, *Graphic techniques for exploring social network data*. New York: Cambridge University Press, 2005, pages 248-269 in 'Models and methods in social network analysis' edited by P.J. Carrington, J. Scott, and S. Wasserman.

[10]. O. Frank and D. Strauss, "Markov graphs," *Journal of the American Statistical Association*, vol. 81, pp. 832–842, 1986.

[11]. S. Wasserman and P. Pattison, "Logit models and logistic regression for social networks: I. an introduction to markov graphs and *p!*," *Psychometrika*, vol. 61, pp. 401–425, 1996.

[12]. C. Anderson, S. Wasserman, and B. Crouch, "A *p!*primer: logit models for social networks," *Social Networks*, vol. 21, pp. 37–66, 1999.

[13]. T. A. Snijders, P. E. Pattison, G. L. Robins, and M. S. Handcock, "New specifications for exponential random graph models," 2004.

[14]. Anindita Chatterjee, Himadri Nath Moulick, Dr. Chandan Koner, "Use of image processing through Emotion Detection", International Journal on Applications of Information and Communication Engineering, February 2015

[15]. Luiz Marques Afonso, Renato F. de G. Cerqueira and Clarisse Sieckenius de Souza,"Evaluating application programming interfaces as communication artefacts",IBM Reseach paper,Brazil.

[16]. Ravindra S Hegadi, et.al Experimental Study Of Various Data Masking Techniques with Random Replacement using data volume, IJCSIS ISSN 1947-5500,Vol. 9 No. 8, August 2011. (IC=5)

[17]. Yogish H.K et.al, The Descriptive Study of Knowledge Discovery from Web Usage Mining, IJCSI International Journal of Computer Science Issues, Vol. 8, Issue 5, No 1, September 2011 ISSN (Online): 1694-0814, P225-230.

[18]. Manjunath.T.N et.al, A Study on design and analysis of web mart mining and its relevance today-IJEST, ISSN: 0975-5462, Vol. 3 No. 4 Apr 2011.

[19]. Umesh I.M et.al, Cross Industry Survey on Data mining Applications, International Journal of Computer Science and Information Technologies, Vol. 30, 2011, 624-628. (IC=5)