

Evaluation metrics in ontology modules

Zubeida Casmod Khan^{1,2}

¹ Department of Computer Science, University of Cape Town, South Africa

² Council for Scientific and Industrial Research, Pretoria, South Africa
zkhan@csir.co.za

Abstract. In recent years, there has been much advancement in the field of ontology modularity. However, work on evaluation metrics for modules is lacking. For the studies that do exist on evaluation metrics, they specialise on only a few metrics, and there is not always a quantitative approach to calculate them. Furthermore, the metrics are not comprehensive enough to apply to the different types of modules and it is unclear which metrics fare well with, say locality-based modules as opposed to those that fare well with partition modules. In this paper, we create a comprehensive list of both existing and new evaluation metrics for modules, together with equations to measure them. In order to test these metrics with existing modules, we create an ontology module metrics tool, Tool for Ontology Module Metrics (TOMM), to measure the metrics of an ontology module or a set of related ontology modules. This leads to uncovering metrics fare well with which module types.

1 Introduction

A number of methods for ontology modularisation have been proposed in recent years. It is unclear how the quality of an ontology module could be measured. While there are few studies on evaluation ontology modules, they focus on a few of the metrics, such as size, cohesion, coupling, correctness, and completeness [12, 13, 17], and these metrics are not comprehensive enough to apply to the variety of types of modules that exist. Another problem concerning the evaluation metrics is that while many of them are described in several works, there is no formula designed to measure them. For instance intra-module distance, to measure the distance between entities in a module. Furthermore, metrics such as size do not fare well with modules created using locality-based techniques, while completeness and correctness do not measure well with partition-based modules. This could mean that only specific metrics must be used to acquire meaningful results about the quality of an ontology module, based on the nature of the module. To date, there is no work done on linking evaluation metrics for modules, to other characteristics of modules.

To solve these problems, we look at existing modules, from ontology repositories and the literature to determine whether there are metrics that are lacking. Thereafter, we create a comprehensive list of both existing and new evaluation metrics, and create equations to quantitatively measure them. Lastly, we developed software support, Tool for Ontology Modularity Metrics (TOMM), to apply

them to modules to measure these metrics. We have applied TOMM to a set of diverse ontology modules, and analysed the data gaining insight into the expected values for evaluation metrics for the different types of modules. TOMM can be downloaded from <http://www.thezfiles.co.za/Modularity/TOMM.zip>.

The remainder of the paper is structured as followed. A related works is conducted in Section 2, followed by the list of evaluation metrics for modules in Section 3. The software, TOMM, and experimental evaluation is presented in Section 4, and a discussion in Section 5. Lastly, we conclude in Section 6.

2 Related works

The framework for ontology modularity [7] consists of dimensions for modularity such as use-cases, techniques, types, and properties. The dependencies between dimensions can be used to guide the modularisation process and annotate the modules with metadata. By adding on evaluation metrics as a dimension to the framework, it is possible to reveal new insights concerning the quality of modules.

Pathek et. al [12] identified main properties that modules need to satisfy, such as size, correctness, completeness, and evaluated these using existing tools. It was found that module correctness is satisfied by most techniques. However, completeness and size are difficult to satisfy. The logic-based approaches for modularity tend to result in modules where completeness is achieved. The graph-based approaches generate modules of smaller size that are not logically complete.

Schlicht and Stuckenschmidt created a set of structural criteria for ontology modules [13]. The authors argue that this structural criteria has an effect on efficiency, robustness and maintainability for the application of semantics-based peer-to-peer systems. The structural criteria proposed include connectedness, size, and redundancy of representation. The authors propose quantitative functions which can be used to formally measure each criteria value. SWOOP and PATO modularity tools are then evaluated using these structural criteria. It is found that SWOOP favours modules with a good connectedness, over modules with suitable size values. With PATO, a threshold value could be selected and it is observed that as the threshold value is increased, so is the size suitability of the module, while the connectedness value worsens.

Yao et. al introduce cohesion metrics for ontologies [17]. Cohesion generally refers to the extent to which entities in a module are related. The metrics that they propose for this are: number of root classes, number of leaf classes, and average depth of inheritance tree of all leaf node. These metrics, however, are not aimed at evaluating the quality of modules but are rather general for all ontologies.

3 Evaluation metrics

The list of evaluation metrics for modularity was compiled by studying existing literature on modularity. Metrics that were lacking or not properly defined were then defined and formulated as new metrics.

3.1 Existing evaluation metrics

In this section we provide definitions of the existing evaluation metrics, and equations to measure them, where defined. The metrics with an asterisk are those that do not have defined equations, hence we created our own equations for calculations.

Size Size refers to the number of entities in a module, $|M|$. This can be further subdivided into class size $|C|$, object property size $|OP|$, data property size $|DP|$, and individual size $|I|$. Several existing works mention size as a modularity evaluation criterion [2, 3, 10, 12, 13].

$$Size(M) = |M| = |C| + |OP| + |DP| + |I| \quad (1)$$

Appropriateness of module size Schlicht and Stuckenschmidt define the appropriateness of module size to map the size of an ontology module to some appropriateness values [13]. They propose an appropriate function to measure this. The appropriateness value is between 0 and 1 where a module with an optimal size is of value 1. Schlicht and Stuckenschmidt propose a function, based on software design principles; since the optimal size of software modules is between 200-300 logical lines of software code, Schlicht and Stuckenschmidt use an axiom value of 250 to of the optimal size for an ontology. The proposed function restricts that modules to be between 0 and 500 axioms.

The appropriateness equation by Schlicht and Stuckenschmidt is defined as follows:

$$Appropriate(x) = \frac{1}{2} - \frac{1}{2} \cos(x \cdot \frac{\pi}{250}) \quad (2)$$

where x is the number of axioms in the module.

Intra-module distance* d'Aquin et al. define the intra-module distance in a module as the distance between entities in a module [3]. It is calculated by counting the number of relations in the shortest path from one entity to the other, for every entity in the module.

Based on the above definition by d'Aquin et al., we formulate an equation to measure the intra-module distance of a module that considers the distance between an entity to another in terms of shortest-path relations. For measuring this distance, we use Freeman's Farness value [4]. In the field of network centrality, Freeman's Farness value of a node is described as the sum of its distances to all other nodes in the network.

$$Intra-module\ distance(M) = \sum_i^n \frac{Farness(i)}{2} \quad (3)$$

where n is the number of nodes in the module, and Freeman's Farness value is defined as follows:

$$Farness(i) = \sum_j^n distance_{ij} \quad (4)$$

The distance is measured as the length of the shortest path between entities.

Cohesion Cohesion refers to the extent to which entities in a module are related. Several works describe ontology cohesion as a set of metrics to measure the modular relatedness of ontologies [5, 9, 10, 17]. There are a number of different proposed metrics to measure cohesion such as the number of root classes, number of leaf classes, and average depth of inheritance tree of all leaf nodes [17]. These metrics are, however, not relevant for ontology modules as they consider the ontologies themselves, and not the modules. When an ontology is modularised, some of the relations between entities are disconnected over the modules. This disconnection has an effect on the cohesion of module which is not reflected by these modules. In order to accurately measure the cohesiveness for ontology modules, we use a metric defined by Oh et al. [10].

$$Cohesion(M) = \begin{cases} \sum_{C_i \in M} \sum_{C_j \in M} \frac{SR(c_i, c_j)}{|M|(|M|-1)} & \text{if } |M| > 1 \\ 1 & \text{otherwise} \end{cases} \quad (5)$$

where $|M|$ is the number of entities in the module. The product of $|M|(|M|-1)$ represents the number of possible relations between entities in M . The strength of relation for each entity is calculated based on the farness centrality measure for graph theory proposed by Freeman [4] from equation 4.

$$SR(c_i, c_j) = \begin{cases} \frac{1}{farness(i)} & \text{if relations exist between } c_i \text{ and } c_j \\ 0 & \text{otherwise} \end{cases} \quad (6)$$

Correctness Correctness states that every axiom that exists in the module also exists in the original ontology and that nothing new should be added to the module. Several works mention the logical correctness criterion [1, 3, 8, 12].

$$Correctness(M) = M \subseteq O \quad (7)$$

Completeness In ontology modules, a module is logically complete if the meaning of every entity is preserved as in the source ontology. The completeness property evaluates that for a given set of entities or signature, every axiom that is relevant to the entity as in the source ontology is captured in the module. Several works mention the logical completeness criterion [1, 3, 8, 12].

$$Completeness(M) = \sum_i^n = Axioms(Entity_i(M)) \models Axioms(Entity_i(O)) \quad (8)$$

Inter-module distance* Existing works describe the inter-module distance in a set of modules as the number of modules that have to be considered to relate two entities [2, 3]. Based on this definition, we have created an equation to measure the inter-module distance of a network of modules.

$$\begin{aligned}
 & \textit{Inter-module distance} = \\
 & \begin{cases} \sum_{C_i, C_j \in (M_i, \dots, M_n)} \frac{NM(C_i, C_j)}{|(M_i, \dots, M_n)|(|(M_i, \dots, M_n)| - 1)} & |(M_i, \dots, M_n)| > 1 \\ 1 & \textit{otherwise} \end{cases} \quad (9)
 \end{aligned}$$

where $NM(C_i, C_j)$ is the number of modules to consider to relate entities i and j . The product of $|M_i, \dots, M_n|(|M_i, \dots, M_n| - 1)$ represents the number of possible relations between entities in a set of modules (M_i, \dots, M_n) .

Coupling* Coupling has been defined in several works as a measure of the degree of interdependence of a module [5, 9–11]. The coupling value is high if entities in a module have strong relations to entities in other modules; it is difficult to modify and update such modules independently because they affect other modules in the system.

Therefore, to measure the coupling of a module, we define our own measure as a ratio of the number of external links (axioms) between a module M_i and M_j , NEL_{M_i, M_j} for n modules in a system to every possible external link between a module M_i and M_j in a system.

$$\textit{Coupling}(M_i) = \begin{cases} \sum_{i=0}^n \sum_{\substack{j=0 \\ i \neq j}}^n \frac{NEL_{M_i, M_j}}{|M_i||M_j|} & NEL_{M_i, M_j} > 0 \\ 0 & \textit{otherwise} \end{cases} \quad (10)$$

where $|M_i|$ is the number of entities in the current module and $|M_j|$ is the number of entities in a related module in the set of n modules.

Attribute richness Tartir et al. [14] define this as the average number of attributes per class. Each entity in an ontology has a number of axioms defined describing it. These are referred to as attributes or slots and measure the attribute richness.

$$AR(M) = \frac{|att|}{|C|} \quad (11)$$

where att is the number of attributes of all entities and $|C|$ is the number of classes in the module. In an ontology, an attribute is used to describe an entity and each attribute, or data type, has a name and value.

Inheritance richness Tartir et al. [14] define this as how the knowledge is distributed across the ontology. There is a distinction between an ontology with a vertical structure or an ontology with a flat structure. Ontologies with few inheritance levels are flat or horizontal in structure where classes have a large number of subclasses. On the other hand, ontologies with many inheritance levels have a deep or vertical hierarchy.

$$IR_S(M) = \frac{\sum_{C_i \in C} |H^C(C_1, C_i)|}{|C|} \quad (12)$$

where $|H^C(C_1, C_i)|$ is the number of subclasses per class and $|C|$ is the total number of classes in the ontology.

In this section, we provided an overview of the metrics that exist for measuring the quality of ontology modules. From the list of 12 existing metrics, there were no quantitative equation for measuring the following 3 metrics: intra-module distance, inter-module distance, and coupling, hence we defined our own equations.

3.2 New evaluation metrics

There are a few metrics that have not been defined in existing works. In this section, we define them, together with equations for measuring them.

Relative size We define relative size as the size of the module, ie., number of classes, properties and individuals compared to the original ontology.

We have created an equation to calculate the size of an ontology module as a ratio of entities of the module over the entities of the source ontology as follows.

$$Relative\ size = \frac{|M|}{|O|} \quad (13)$$

where $|M|$ is the number of entities in the module and where $|O|$ is the number of entities in the source ontology.

Atomic Size The notion of atoms within ontology modules was first introduced by Del Vescovo et al. [15] in a study of BioPortal repository ontologies [16]. An atom is a group of axioms within an ontology that have dependencies between each other. The atomic size of a module, is the average size of a group of inter-dependent axioms in a module.

We formulate an equation to measure the atomic size of a module by using the sum of all the atoms present in the module, and the size of the ontology.

$$Atomic\ Size(M) = \sum_i^n \frac{Atom_i}{|M|} \quad (14)$$

Relative Intra-module distance We define the relative intra-module distance of a module as the difference between distances of entities in a module M to a source ontology O . To compare the distances of the original ontology, we compute the farness values for the subset of nodes that exist in a module, which is used to calculate the intra-module distance of the original ontology. The difference between the intra-module distance of the source ontology and the module would reveal if the overall distance between the entities in the module has been reduced, and by how many distance units.

We formulate the relative intra-module distance of a module as follows.

$$\text{Relative intra-module distance}(M) = \frac{\text{Intra-module distance}(O)}{\text{Intra-module distance}(M)} \quad (15)$$

Encapsulation d’Aquin et al. mention encapsulation with the notion that “a module can be easily exchanged for another, or internally modified, without side-effects on the application can be a good indication of the quality of the module” [3]. Given this idea, we define the encapsulation of a module in a set of modules as a measure of knowledge preservation within the given module.

We have designed an equation to calculate the encapsulation of a module in a given set of modules. For a module, with a set of $n - 1$ related modules, we measure the number of axioms that exist in the given module, $|Ax_i|$. For each module that is related to the given module, we measure the number of axioms that exist in the related module but also exist in the given module, $|Ax_{ij}|$.

$$\text{Encapsulation}(M_i) = 1 - \frac{\sum_{j=1}^{n-1} \frac{|Ax_{ij}|}{|Ax_i|}}{n} \quad (16)$$

Encapsulation values in modules that are equal or close to 1 indicates a good encapsulation value; all or most of the knowledge has been encapsulated and privacy has been completely preserved. Conversely, values that are equal to or close to 0 indicates a poor encapsulation value; none or very little of the knowledge has been encapsulated and privacy has not been preserved.

Independence Independence evaluates whether a module is self-contained and can be updated and reused separately. In this way, ontology modules can evolve independently. Thus, the semantics of the entire ontology could change without the need for all the modules to be changed.

In order to determine whether a module is independent, we use two metrics, ie, the encapsulation and the coupling measure. Recall that encapsulation values of 1 indicate that all of the knowledge is preserved in the module. This means that knowledge is not shared among modules hence updates to the module do not affect the entire system. A low coupling value of 0 means that the module does not share any relations with external modules. Hence a module is set to be independent if it has an encapsulation value of 1 and a coupling value of 0. This can be checked using the following code snippet.

```

if  $Encapsulation(M) == 1$  and  $Coupling(M) == 0$  then
   $M \leftarrow independent$ 
else
   $module \leftarrow not\ independent$ 
end if

```

4 Tool for ontology modularity metrics

We have created TOMM to evaluate ontology modules which will lead to uncovering problems with modularity evaluation. TOMM allows a user to upload an ontology module, or a set of related ontology modules, together with an original ontology (if it exists), and calculates the metrics for each ontology module. The screenshot for the interface of TOMM is displayed in Fig. 1. The metrics are then saved as a text file on the user's computer, as shown in Fig. 2.

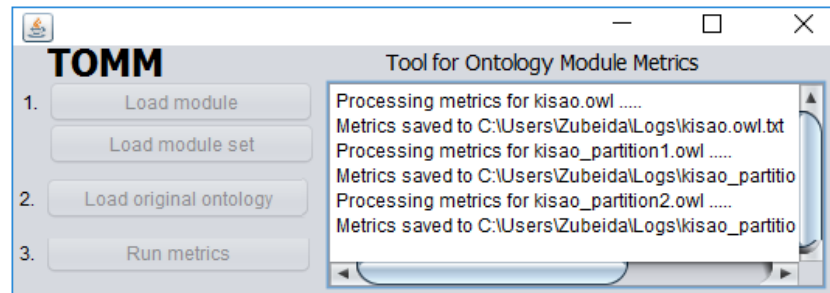


Fig. 1. The interface of TOMM.

```

Log file for aminoacid partition2.owl Attribute richness of ontology: 0.0
Metrics for aminoacid_partition2.owl Inheritance richness of ontology: 0.0
No. of classes in ontology: 18 Encapsulation of ontology 1.0
No. of OP in ontology: 0 Coupling of ontology 0.0
No. of DP in ontology: 0 Is the ontology independent? true
No. of Ind in ontology: 0
Size of ontology: 18
No. of axioms in ontology: 49
Appropriateness of ontology: 0.0918
3037464140803
Intra module distance: 225.5
Cohesion of ontology: 0.2050653594
7712437
Relative Metrics for aminoacid partition2
.owl compared to amino acid - inferred.owl
Relative size of module: 0.34615384615384615
Relative intra module distance of module:
3.450110864745011
The module is logically correct,
no new axioms have been added to the ontology.
The module is logically complete.

```

Fig. 2. A log of the metrics generated with TOMM for an ontology module.

4.1 Experimental evaluation

The purpose of the experimental evaluation is to determine which metrics fare well with which module types to measure the quality of an ontology module.

Materials and methods The method for the experiment is as follows:

1. Collect the set of ontology modules from the test files for existing framework for modularity [7].
2. Run the TOMM metric tool for each module.
3. Conduct a statistical analysis from the evaluation metric results for each module.

The materials used for the experiment were as follows: Protégé v4.3 ([6]), TOMM, and a set of ontology modules. The sample size was 189 different types of ontology modules. All these test files used for this experimental evaluation can be downloaded from www.thezfiles.co.za/Modules/testfiles.zip.

There are 14 types of modules in the set, defined in the ontology modularity framework. We briefly describe them here; a detailed description with examples for each module type and the framework is presented elsewhere [7].

- T1 *Ontology design pattern modules* An ontology is modularised by identifying a part of the ontology for general reuse.
- T2 *Subject domain modules* A large domain is divided by subdomains present in the ontology.
- T3 *Isolation branch modules* A subset of entities from an ontology is extracted but entities with weak dependencies to the signature are not to be included in the module.
- T4 *Locality modules* A subset of entities from an ontology is extracted, including all entities that are dependent on the subset.
- T5 *Privacy modules* Some information is hidden from an ontology.
- T6 *Domain coverage modules* A large ontology is partitioned by its graphical structure and placement of entities in the taxonomy.
- T7 *Ontology matching modules* An ontology is modularised for ontology matching into disjoint modules so that there is no repetition of entities.
- T8 *Optimal reasoning modules* An ontology is split into smaller modules to aid in overall reasoning over the ontology.
- T9 *Axiom abstraction modules* An ontology is modularised to have fewer axioms with object properties relating classes, to decrease the horizontal structure of the ontology.
- T10 *Entity type abstraction modules* An ontology is modularised by removing a certain type of entity e.g., data properties or object properties.
- T11 *High-level abstraction modules* An ontology is modularised by removing lower-level classes and only keeping higher-level classes.
- T12 *Weighted modules* An ontology is modularised by a weighting decided by the developer.
- T13 *Expressiveness sub-language modules* An ontology is modularised by using a sub-language of a core ontology language.
- T14 *Expressiveness feature modules* An ontology is modularised by using limited language features.

Results We ran TOMM for each of the 189 modules of the test files. Metrics were successfully generated for 188 modules; the ‘FMA_subset’ module was still too large for TOMM to process. We have analysed those metrics pertaining to structure (size, cohesion etc.) which is displayed in Table 1. The table indicates which modules fare well with the structural metrics.

Table 1. Averages for the structural metrics of the set of modules; app. =appropriateness and coh. =cohesion.

	Size	Relative Size	No. of axioms	App.	Atomic size	Intramodule distance	Relative intramodule distance	Coh.
T1	10.85	0.06	39.31	0.07	35.35	2.42	29.00	0.054
T2	61.31	-	264.86	0.23	5.19	7896.38	-	0.03
T3	84.86	0.78	363.14	0.21	6.26	1839.57	3.43	0.07
T4	29.00	0.34	261.67	0.47	10.10	332.00	12583.36	0.06
T5	33.50	0.30	168.50	0.61	7.20	357.00	72.41	0.08
T6	417.20	0.21	922.5	0.49	3.17	252386.95	2434.84	0.13
T7	2.26	0.02	14.02	0.009	1.33	0.48	9196.98	0.15
T8	840.50	0.60	2157.75	-	3.77	80118.88	1309.34	0.003
T9	94.00	1.00	884.00	-	2.89	6161.00	1.00	0.07
T10	103.00	-	257.00	0.99	4.21	11798.00	-	0.02
T11	278.33	0.51	712.67	0.89	3.72	324.33	1532.01	0.002
T12	158.00	0.41	580.67	0.02	5.84	2309.33	21.61	0.02
T13	301.00	1.00	991.83	0.65	4.35	116466.67	1.00	0.01
T14	1356.00	0.97	4360.00	-	5.57	697630.50	1.00	0.01

Modules T3 (isolation branch), T8 (optimal reasoning), T9 (axiom abstraction), T11 (high-level abstraction), T13 (expressiveness sub-language), and T14 (expressiveness feature), are all as large as over 50% of the original ontologies, according to the relative size metrics. T1 (ontology design pattern) is very small, less than 1% compared to the original ontology. T2 (subject domain) and T10 (entity type abstraction) could not be evaluated with the relative size metric as there are no original ontologies.

For the appropriateness value, which is optimally close to a value of 1, module T10 (entity type abstraction) performs best with an average of 0.99. Modules T8 (optimal reasoning), T9 (axiom abstraction), and T14 (expressiveness feature) all contain more than 500 axioms hence could not be evaluated with appropriateness values. For the atomic size values, T4 (locality) is the largest at 10.1, and T7 (ontology matching) is the lowest at 1.33. T8 (optimal reasoning), T9 (axiom abstraction), T10 (entity type abstraction), and T14 (expressiveness feature) cannot be evaluated with the appropriateness values as none of its modules are within the range of 0-500 axioms which is required for measuring a module’s appropriateness.

The relative intra-module distance values determine how much the module has been reduced. T4 (locality) has the highest value of 12583.36. T2 (subject

domain) and T10 (axiom abstraction) cannot be evaluated with this measure as there are no original ontologies. For the cohesion values, the extent of which entities in a module are related, all the modules have low values; T6 (domain coverage) and T7 (ontology matching) have the highest values of 0.13 and 0.15 respectively.

5 Discussion

The list of module metrics that was compiled is a first step in solving ontology developers' problems regarding the evaluation of ontology modules. The metrics that are programmed into TOMM, allows one to evaluate ontology modules using a variety of metrics such as logical aspects (completeness and correctness), structural aspects (size), relational aspects (coupling), etc.

We have evaluated a set of modules with TOMM, and analysed the metrics pertaining to the structural aspects of the module. The results reveal which metrics work well with which type of module. For instance T1 (ontology design patterns) modules are relatively small compared to the original ontology. The remaining metrics such as coupling, correctness etc. need to be analysed together with the set of modules. If these metrics are linked to other characteristics from the framework for ontology modularity [7], it would be possible to determine which metrics fare well with which use-cases, techniques, and properties for modules.

6 Conclusion

The lack of work on metrics for ontology modules causes problems for ontology developers for the evaluation of modules. In this paper, we compiled a list of existing and new metrics for evaluating modules, together with formal definitions and equations for computation. This was then programmed into TOMM, for ontology developers to evaluate their ontologies. An experimental evaluation with a set of ontology modules was then conducted. The structural metrics were analysed for the set of modules revealing which metrics work well with which types of modules.

For future work, we aim to achieve more insight into module evaluation by linking the module evaluation metrics to other characteristics of the ontology modularity framework.

References

1. Cuenca Grau, B., Parsia, B., Sirin, E., Kalyanpur, A.: Modularity and Web Ontologies. In: 10th International Conference on Principles of Knowledge Representation and Reasoning (KR'06). pp. 198–209. AAAI Press (2006), june 2-5, Lake District, United Kingdom.

2. d'Aquin, M., Schlicht, A., Stuckenschmidt, H., Sabou, M.: Ontology modularization for knowledge selection: Experiments and evaluations. In: Wagner, R., Revell, N., Pernul, G. (eds.) *Proceedings of the 18th International Conference on Database and Expert Systems Applications (DEXA'07)*. Lecture Notes in Computer Science, vol. 4653, pp. 874–883. Springer (2007), regensburg, Germany, September 3-7, 2007
3. d'Aquin, M., Schlicht, A., Stuckenschmidt, H., Sabou, M.: Criteria and evaluation for ontology modularization techniques. In: *Modular Ontologies: Concepts, Theories and Techniques for Knowledge Modularization*, Lecture Notes in Computer Science, vol. 5445, pp. 67–89. Springer (2009)
4. Freeman, L.C.: Centrality in social networks conceptual clarification. *Social Networks* 1(3), 215–239 (1978)
5. García, J., Peñalvo, F.J.G., Therón, R.: A survey on ontology metrics. In: Lytras, M.D., de Pablos, P.O., Ziderman, A., Roulstone, A., Maurer, H.A., Imber, J.B. (eds.) *Third World Summit on the Knowledge Society, (WSKS'10)*. Communications in Computer and Information Science, vol. 111, pp. 22–27. Springer (2010), corfu, Greece, September 22-24
6. Gennari, J.H., Musen, M.A., Fergerson, R.W., Grosso, W.E., Crubézy, M., Eriksson, H., Noy, N.F., Tu, S.W.: The evolution of Protégé: an environment for knowledge-based systems development. *International Journal of Human-Computer Studies* 58(1), 89–123 (2003)
7. Khan, Z.C., Keet, C.M.: An empirically-based framework for ontology modularisation. *Applied Ontology* 10(3-4), 171–195 (2015), <http://dx.doi.org/10.3233/AO-150151>
8. Loebe, F.: Requirements for logical modules. In: Haase, P., Honavar, V., Kutz, O., Sure, Y., Tamin, A. (eds.) *The 1st International Workshop on Modular Ontologies (WoMO'06)*. CEUR Workshop Proceedings, vol. 232. CEUR-WS.org (2006), iSWC'06 November 5, Athens, Georgia, USA
9. Oh, S., Ahn, J.: Ontology module metrics. In: *International Conference on e-Business Engineering, (ICEBE'09)*. pp. 11–18. IEEE Computer Society (2009), macau, China, 21-23 October
10. Oh, S., Yeom, H.Y., Ahn, J.: Cohesion and coupling metrics for ontology modules. *Information Technology and Management* 12(2), 81–96 (2011)
11. Orme, A.M., Yao, H., Eitzkorn, L.H.: Coupling metrics for ontology-based systems. *IEEE Software* 23(2), 102–108 (2006)
12. Pathak, J., Johnson, T.M., Chute, C.G.: Survey of modular ontology techniques and their applications in the biomedical domain. *Integrated Computer-Aided Engineering* 16(3), 225–242 (2009)
13. Schlicht, A., Stuckenschmidt, H.: Towards structural criteria for ontology modularization. In: *Proceedings of the 1st International Workshop on Modular Ontologies, (WoMO'06)*. CEUR Workshop Proceedings, vol. 232. CEUR-WS.org (2006), iSWC'06 November 5, 2006, Athens, Georgia, USA
14. Tartir, S., Arpinar, I.B., Moore, M., Sheth, A.P., Aleman-Meza, B.: OntoQA: Metric-based ontology quality analysis. *IEEE Workshop on Knowledge Acquisition from Distributed, Autonomous, Semantically Heterogeneous Data and Knowledge Sources* 9 (2005)
15. Vescovo, C.D., Gessler, D., Klinov, P., Parsia, B., Sattler, U., Schneider, T., Winget, A.: Decomposition and Modular Structure of BioPortal Ontologies. In: *10th International Conference on The International Semantic Web Conference (ISWC'10)*. Lecture Notes in Computer Science, vol. 7031, pp. 130–145. Springer (2011), october 23-27, Bonn, Germany.

16. Whetzel, P.L., Noy, N.F., Shah, N.H., Alexander, P.R., Nyulas, C., Tudorache, T., Musen, M.A.: BioPortal: enhanced functionality via new web services from the national center for biomedical ontology to access and use ontologies in software applications. *Nucleic Acids Research* 39(Web-Server-Issue) (2011)
17. Yao, H., Orme, A.M., Etkorn, L.: Cohesion metrics for ontology design and application. *Journal of Computer science* 1(1), 107 (2005)