A software processing chain for evaluating thesaurus quality

Javier Lacasta*, Gilles Falquet†, Javier Nogueras-Iso*, and Javier Zarazaga-Soria*

* Computer Science and Systems Engineering Dept., Universidad de Zaragoza, Spain.
† Centre Universitaire d'Informatique, Université de Genève, Switzerland.

Abstract. Thesauri are knowledge models commonly used for information classification and retrieval whose structure is defined by standards that describe the main features the concepts and relations must have. However, following these standards requires a deep knowledge of the field the thesaurus is going to cover and experience in their creation. To help in this task, this paper describes a software processing chain that provides different validation components that evaluates the quality of the main thesaurus features.

Keywords: Thesaurus, Digital libraries, Information retrieval, Thesaurus quality, Ontology alignment

1 Introduction

The resources in metadata repositories are frequently classified using thesauri because of their simple structure, the established standards [7] and the integrated support provided by most catalog tools. Keyword based search is the standard for performing searches in many information systems and thesauri are one of the most used models to organize and relate the keywords [19]. The construction of a thesaurus requires a careful selection of the concepts in an area of knowledge and their interrelations in an appropriate general-to-specific hierarchy [5]. However, many factors, such as the lack of experience, costs savings, or the over-adaptation to a data collection, produce models with heterogeneous concepts and relations [4].

Simple edition tools for thesauri focus on providing a suitable environment for the creators to define concepts and relations in a collaborative way. In these tools, quality control is mainly focused on the definition of a human-oriented process were an editor reviews the work previous to its final inclusion in the thesaurus. This approach is especially valid for small thesaurus in which a person can maintain the control over the entire model. However, as the thesaurus size grows, this process becomes more difficult and problems of terminological heterogeneity, overload of specificity, lexical issues in concept labels or unclear hierarchies become common [16, 4].

In a previous work [10], we described a process to detect issues in the thesaurus according to ISO 25964 specification [7] at all different levels (labels,

concepts, and relations). This paper continues in this line of work, and describes the software processing chain created to implement the validation tasks, and the integration framework used to merge these components into a complete validation tool. Instead of a monolithic analysis process, we have opted for using a modular approach in which each library component analyses a single feature. This greatly increases the flexibility of use of the library and facilitates its use in contexts where not all the thesaurus features are required. For example, to perform real-time validation of property values when they are defined in a thesaurus edition tool. Additionally, since some of the validation tasks are intensive in processing, the validation time can be quite considerable. Therefore, in addition to a modular approach, we have used a framework that greatly simplifies the parallel execution of the validation tasks in a single machine or cluster.

The paper is structured as follows. Section 2 summarizes the quality features for which we have created validation components, and reviews existent software solutions. Section 3 describes these components and how they are integrated in a tool. The paper ends with some conclusions and an outlook on future work.

2 Thesaurus quality features and existent quality analysis tools

"The quality" is a measure of excellence or a state of being free from defects, deficiencies and significant variations. ISO 8402 [8] defines the quality as "the totality of features and characteristics of a product or service that bears its ability to satisfy stated or implied needs".

The main sources to identify the quality features of a thesaurus are the existing construction guidelines. They range from practice manuals such as Aitchison et al. [1], to the current international standard ISO 25964 [7]. Pinto [13], Kless and Milton [9], and Mader and Haslhofer [11] are the principal studies focused on identifying the features that determine the quality of thesauri. They focus on concepts, terms, structure and documentation parts, and describe features that need to be reviewed at each level. This information has been compiled from specifications, previous works in the area and user surveys.

From the features described in these works, we have developed components to automatically analyze the main elements described in ISO 25964:

Property completeness measures: These measures are focused on the identification of lacking properties. We analyse the completeness and uniqueness of preferred labels and completeness of definitions.

Property content measures: Their objective is to locate invalid values inside labels. We focus on detecting non-alphabetic characters, adverbs, initial articles, and acronyms (in preferred labels).

Property context measures: These are focused on identifying anomalies involving several labels. This includes detecting duplicated labels and inconsistencies in the use of uppercase and plurals.

Property complexity measures: They provide a measure of the syntactic complexity of the labels, in terms of the use of prepositions, conjunctions and adjectives.

Relation coherence measures: They indicate if the relations are complete, coherent, and semantically correct. RT (Related Term) analysis focuses on detecting non-informative relations (they link hierarchically related concepts). BT/NT (Broader/Narrower Terms) analysis searches for cycles in the model,unlinked concepts and relations that do not associate a superordinate with a subordinate concept. According to ISO 25964, the superordinate must represent a class or whole and subordinate its members or parts.

There are some validation tools such as Mader and Haslhofer [11], Suominen and Mader [17], Eckert [3] or Poveda-Villalón et al. [14] that analyze some of these features, but not all of them. They focus on reviewing the completeness of the properties and relations, and existence of cycles. Only Eckert [3] provides a method to detect issues in the relations, but at the cost of using a collection classified with the thesaurus. Our work deals with all of them, this includes those related to the syntaxis and semantics of the labels, concepts and relations (e.g., the use of adverbs, acronyms, and the meaning of BT/NT relations). The developed components analyze the thesaurus features through structural, lexical, syntactical and semantical checks that validate the labels describing the concepts and the provided relations.

3 Design of the software processing chain

For the development of the software that performs the validation of the previously described features, we have opted for a modular approach in which each feature is reviewed by a different class in the validation library. This provides a great flexibility in terms of adjusting the system to different needs, such as the development of tools that only perform a subset of the implemented validation tasks, and the addition or improvement of functionality.

To implement these modules we have used the Spring framework[1]. This framework simplifies the use of the dependency-injection pattern, allowing the definition of data flows between completely decoupled components through a configuration file (or even class annotations). These data flows can be defined so the unrelated parts can be automatically executed in parallel by different processes in the same or in different computers without the programmer having to program the distribution and aggregation of the data, or the synchronization of the processes. Additionally, it provides other useful functionality such as transaction control to deal with errors, and logging (between many others).

Some of the required validation tasks are quite intensive in processing due to their complexity and/or the amount of data the thesaurus contains. This makes difficult the construction of a fast validation tool that quickly performs the analysis. In this context, the use of Spring has facilitated us the construction

[1] https://spring.io/

of a parallel execution flow for unrelated validation tasks. Additionally, some of these tasks are composed of independent sub-processes, which are also independent between them. From a technical point of view, the validation tasks can be classified in three families: those that require the processing of some properties distributed along the whole thesaurus, those that require the analysis of several properties of each concept, and those that analyze the content of the instances of a single property. From them, only those analysis affecting the whole set of properties need to be executed as a whole. Those affecting a concept or a property can be divided in as many concepts or properties the thesaurus contains.

From a very general point of view, the validation tool is composed of a reader, a validator, and a report generator (see Figure 1). The reader provides the access to the repository and loads the thesaurus in memory. The validator provides the data flow that decomposes the thesaurus and provides them to the different validation sub-components. Finally, the report generator creates a human friendly document describing the detected issues. The current implementation of the reader is focused on loading thesaurus in SKOS format [12] and the report generator provides a simple textual report. However, thanks to the development framework used, these components can be replaced by other ones able to load and generate other formats (e.g., HTML, Excel, PDF, and so on) with a simple modification of the process configuration file.

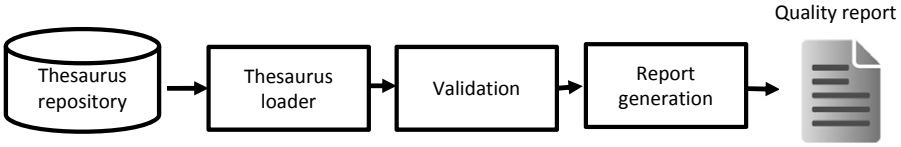

Fig. 1. Thesaurus validation process

The subset of the validation tasks that require the analysis of the whole thesaurus are the detection of non "Informative RT" relations, "BT/NT cycles" and "duplicated labels" (see Figure 2). RT analysis involves the processing of all the other concepts in the same branch to detect if the are already related by a BT/NT relation. Cycles are located using a modified version of Tarjan's strongly connected components algorithm [18] that identifies the relation that generates the cycle (it points out to a broader concept). With respect to duplicated labels, it is needed to compare each label with the rest, but it has to be done aggregating them per language, since different languages can use the same word to describe a concept. Therefore, the validation task has been implemented to be independent of the language, and the framework has been configured to call it as many times as languages are in the thesaurus.

The validation tasks centered in detecting properties whose context is a single thesaurus concept are the detection of the completeness in the "definitions", "preferred labels" (there must be only one per concept and language) and "BT/NT relations" (no orphan concepts or branches). The data flow is de-

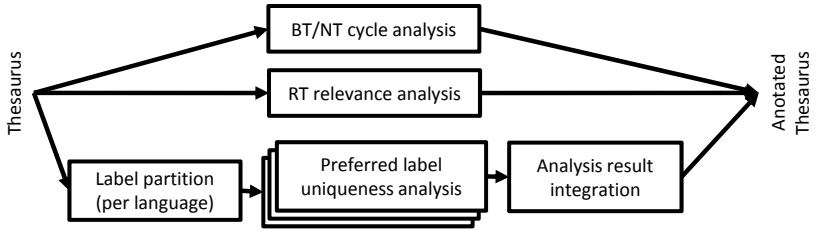

Fig. 2. Thesaurus level validation tasks

picted in Figure 3. The algorithms used for these tasks are unremarkable, as they only check the existence of the properties and make annotations to the concept that are latter used in the report generation step.

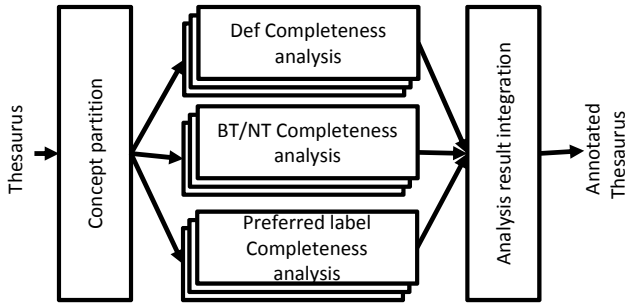

Fig. 3. Concept level validation tasks

The set of tasks focused on properties of a single label (preferred or alternative ones) includes some trivial and some complex tasks (see Figure 4). In any case, an important characteristic all of them have is their dependence on the language. Therefore, is needed to define a new family of processors adapted to each language that implement all the tasks in this category.

Among the trivial tasks we found the identification of "non-alphabetic characters", "acronyms", "initial uppercase", and "plurals". The first three only require simple character comparisons, while plural detection has required the use of an adapted version of the Solr minimal stemmer [15] that detects plurals instead of removing them. The use of uppercase and plural in a label is not per se a quality feature, but the homogeneous use along all the thesaurus is. Therefore, in this phase, uppercase and plural labels are tagged so the real quality features can be analyzed latter in a simpler way.

The rest of the tasks perform a syntactic analysis to the label being processed. For that purpose, we have used GATE [2], a software library for natural language processing that provides part of speech tagging functionality. With the labels properly tagged, the identification of "conjunctions", "adverbs", "initial

articles", "prepositional phrases" (only for English), and "too long and complex phrases" (detected by counting adjectives) only requires the revision of the generated part of speech tags. In addition to the syntactic analysis, an additional task aligns the labels to WordNet, a lexical database originally in English that groups nouns, verbs, adjectives and adverbs into sets of cognitive synonyms (Synsets), each expressing a distinct concept. Synsets are interlinked by means of conceptual-semantic and lexical relations providing a hypernym/hyponym hierarchy of semantically related concepts. This is done as a previous step for the identification of the semantic correctness of BT/NT relations. Since the thesaurus labels can be in languages different from English, instead of the pure WordNet, we have used the Open Multilingual WordNet (OMWordNet)[2]. This is an extension of WordNet that maintains the concept relation structure but incorporating labels in several additional languages. In this step, a direct lexical alignment is performed, annotating the label with all the WordNet concepts that share the label (ignoring case and plural).

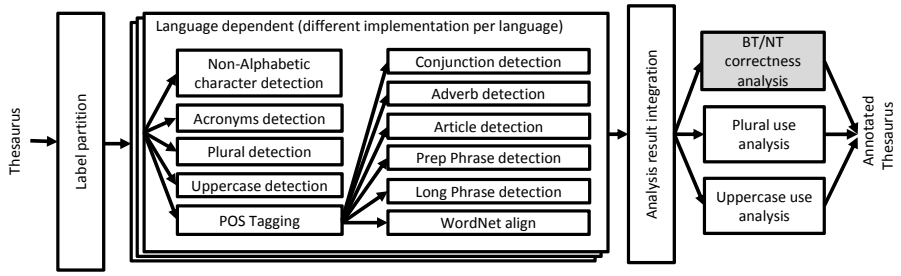

Fig. 4. Label level validation tasks

Once the plural, initial uppercase tags, and WordNet senses are added at each label, a last set of validation tasks that require this information as prerequisite is executed. These tasks are the check of a "homogeneous use of plurals and initial uppercase in labels", and the "semantic correctness of BT/NT relations". Plural and uppercase check just requires counting the occurrences of each feature and marking as incorrect those that are different from the majority. With respect to the detection of the correctness of BT/NT relations, it aligns each concept with the DOLCE ontology [6] to identify the semantic meaning of the relations and therefore identify the incorrect ones. DOLCE provides top level categories of concepts with a deep semantic net of relations between them. Some of these relations, such as "participant'" or "exact location of" intrinsically provide a superordinate and subordinate meaning and they can be considered as BT/NT specializations. These concepts are too generic, so a direct alignment with the thesaurus is not possible. Therefore, the alignment with DOLCE is done thanks to the previously obtained alignment with WordNet and a manual alignment

[2] http://compling.hss.ntu.edu.sg/omw/

between WordNet Synsets and DOLCE categories [6]. However, to be able to use the alignment, it is needed to select the correct meaning of each thesaurus in WordNet from the multiple ones provided by each language. This is done by selecting the most common Synset between the obtained for all the labels in different languages of a concept. Additionally, in the case where there are several alternatives with the same occurrences, the process uses previously established alignments of other concepts in the branch as the disambiguation context. In this case, the semantically closest sense in WordNet to those already selected for other concepts is the chosen one.

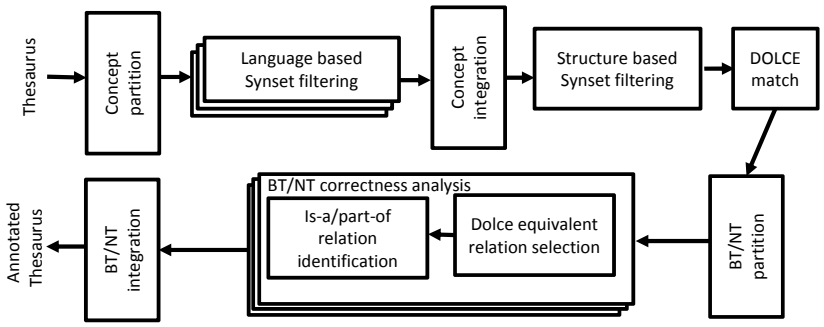

Fig. 5. Detail of the BT/NT semantic correctness validation task

The final step in this task reviews the concepts involved in each BT/NT relation to obtain the equivalent relation in DOLCE. We have identified three families of DOLCE relations (subclass, participation and location) that are compatible with BT/NT semantics. Therefore, when one relation in this family is found, the original BT/NT is tagged as correct, in other case is considered incorrect. A detailed description of each family is described next:

- The subclass relation indicates that the original concepts belong to hierarchically related categories. This does not ensure that the original BT/NT is correct, but because the thesaurus objective is to create generic to specific models, it is a good clue in that direction.
- The participation relation holds between perdurants (activities) and endurants (objects). It indicates elements that are part of an activity, which is a valid BT/NT meaning (e.g., horse piece is *part-of* a chess game or a car is *part-of* a car accident).
- With respect to the location related properties (from spatial to conceptual location), they may provide a *part-of* meaning (e.g., fountain *part-of* park) or an *is-a* meaning (e.g., linear town *is-a* kind of urban morphology), both valid in the thesaurus context. If a BT/NT relation is assigned to one of these families, we consider it as correct.

The results of all these tasks are provided to the report generator that generates a textual file with a summary of the issues found. The quality of most of the properties is measured with the percentage of correctness of each of the analyzed features. This percentage is calculated in base to the number of properties/concepts/relations analyzed in a feature and the number of errors identified. Only the cycles in the thesaurus are counted with absolute numbers, as they are critical errors in the definition of the thesaurus where a percentage measure of correctness has no sense. It additionally generates additional log files with the list of errors identified. These logs are adapted to each analyzed feature, and identify the concept(s) involved in the erroneous element, and the value of the erroneous property (or relation, depending on the case).

4 Conclusions and future work

This paper has described the library components developed to validate the correctness of the main features of a thesaurus. These components have been used to construct a complete validation tool, but they can be used independently or arranged in other aggregation ways to analyse in other contexts a subset of the quality features identified. The components have been defined as decoupled as it has been possible, in order to allow the tool to be easily extended, reconfigured and parallelized.

Future work will be devoted to improve the current system to provide a public web validation system. This should be quite simple as it would only require the replacement of the components in charge of the thesaurus load and the validation report generation. Additionally, we want to use the validation components in a separate way to integrate them in a thesaurus edition tool, so that the thesaurus features can be validated as soon as a new element is added to the thesaurus. Finally, we want to continue to extend the validation components so that they can be used in other contexts apart from thesaurus analysis. For example, we would like to use our proposed processing chain for analyzing the quality of properties and relations in ontologies.

Acknowledgements

This work has been partially supported by the Keystone COST Action IC1302 and by the University of Zaragoza (project UZ2016-TEC-05).

References

[1] Aitchison, J., Bawden, D., Gilchrist, A.: Thesaurus Construction and Use: A Practical Manual. Routledge (2000)

[2] Cunningham, H., Maynard, D., Bontcheva, K., Tablan, V.: GATE: an architecture for development of robust HLT applications. In: Proceedings of the 40th annual meeting on association for computational linguistics. pp. 168–175. Association for Computational Linguistics (2002)

[3] Eckert, K.: Usage-driven Maintenance of Knowledge Organization Systems. Ph.D. thesis, Universitat Mannheim (2012)
[4] Fischer, D.H.: From thesauri towards ontologies? In: Structures and relations in knowledge organization - 5th International ISKO Conference. pp. 18–30. Lille (France) (August 1998)
[5] Frakes, W.B., Baeza-Yates, R. (eds.): Information Retrieval: Data Structures & Algorithms, chap. Thesaurus construction, pp. 161–218. Addison Wesley (1992)
[6] Gangemi, A., Guarino, N., Masolo, C., Oltramari, A.: Sweetening WORDNET with DOLCE. AI Magazine 24(3), 13–24 (September 2003)
[7] International Organization for Standardization: Thesauri and interoperability with other vocabularies. ISO 25694, International Organization for Standardization (ISO) (2011)
[8] International Organization for Standarization: Quality management and quality assurance. ISO 8402, International Organization for Standarization (1994)
[9] Kless, D., Milton, S.: Towards quality measures for evaluating thesauri. Communications in Computer and Information Science. Metadata and Semantic Research 108, 312–319 (2010)
[10] Lacasta, J., Falquet, G., Zarazaga-Soria, F.J., Nogueras-Iso, J.: An automatic method for reporting the quality of thesauri. Data & Knowledge Engineering (in press) (2016)
[11] Mader, C., Haslhofer, B.: Perception and relevance of quality issues in web vocabularies. In: I-SEMANTICS'13 Proceedings of the 9th International Conference on Semantic Systems (2013)
[12] Miles, A., Bechhofer, S.: SKOS Simple Knowledge Organization System Reference. No. January in W3C Candidate Recommendation, W3C (2009)
[13] Pinto, M.: A user view of the factors affecting quality of thesauri in social science databases. Library & Information Science Research 30(3), 216–221 (2008)
[14] Poveda-Villalón, M., Suárez-Figueroa, M.C., Gómez-Pérez, A.: Validating ontologies with OOPS! In: 18th International Conference EKAW. Lecture Notes in Computer Science, vol. 7603, pp. 267–281. Springer (2012)
[15] Savoy, J.: Report on CLEF-2001 experiments. Tech. rep., Institut interfacultaire d'informatique, Université de Neuchtel, Switzerland (2001)
[16] Soergel, D.: Indexing Languages and Thesauri: Construction and Maintenance. Melville Pub. Company (1974)
[17] Suominen, O., Mader, C.: Assessing and improving the quality of SKOS vocabularies. Journal on Data Semantics 3(1), 47–73 (2014)
[18] Tarjan, R.E.: Depth-first search and linear graph algorithms. SIAM Journal on Computing 1(2), 146–160 (1972)
[19] Wielemaker, J., Hildebrand, M., van Ossenbruggen, J., Schreiber, G.: Thesaurus-based search in large heterogeneous collections. In: The Semantic Web - ISWC. Lecture Notes in Computer Science, vol. 5318, pp. 695–708 (2008)