Simple Method for Ontology Automatic Extraction from Documents

Andreia Dal Ponte Novelli
Dept. of Computer Science
Aeronautic Technological Institute
Dept. of Informatics
Federal Institute of Sao Paulo
Sao Paulo – Brazil

José Maria Parente de Oliveira
Dept. of Computer Science
Aeronautic Technological Institute
Sao Paulo - Brazil

Abstract—**There are many situations where it is needed to represent and analyze the concepts that describe a document or a collection of documents. One of such situations is the information retrieval, which is becoming more complex by the growing number and variety of document types. One way to represent the concepts is through a formal structure using ontologies. Thus, this article presents a fast and simple method for automatic extraction of ontologies from documents or from a collections of documents that is independent of the document type and uses the junction of several theories and techniques, such as latent semantic for the extraction of initial concepts, Wordnet and similarity to obtain the correlation between the concepts.**

Keywords-document ontology; ontology creation; ontology extraction; concept representation.

I. Introduction

The continuous increase in the amount of documents produced both on the Web and in local repositories makes it increasingly complex and costly to analyze, categorize and retrieve documents without considering the semantics of the whole or each document. Generally, the semantics is analyzed based on the concepts contained in the documents, and the ontologies are one of the ways to represent these concepts. Ontology can be defined as a formal and explicit specification of shared conceptualization [1]. These can also be seen as conceptual models that capture and explain the used vocabulary in semantic specifications. For documents, ontology may be seen as significant group of terms that expresses the vocabulary of the document through concepts and relations modeled after those terms.

Ontology can be constructed in a more general way or a domain-dependent, depending on how general are the sets of concept. In the context of this article, the concepts of ontologies are general, since the terms used for formation of the concept may be present in any document. However, the ontology creation is based on documents from a specific area, thereby resulting ontologies directed to the document domain.

There are many situations where presence of semantics is necessary in order to best perform certain tasks in certain areas, however, depending on the task, it is not necessary that the semantics be extremely detailed regarding the formation of concepts and semantic relations, since the semantics is an auxiliary item to the task. Thus, the proposed method tries to meet the need of creating a simple and meaningful semantic description of documents without analyzing these documents through artificial intelligence techniques, language and context analysis.

The method extracts an ontology from a collection of any documents (text only or structured) or descriptive ontologies of single documents using tools and techniques such as latent semantic analysis, clustering and Wordnet. The initial concepts of the ontology and its relations are obtained from the terms of the documents and other concepts are created from the analysis of the terms using latent semantic and clustering. The relations between the concepts are obtained from analysis using a thesaurus or ontology, and for this work Wordnet was chosen to.

This article is organized as follows: section II presents the state of the art for the automatic extraction of ontologies, in Section III it is presented the concepts of latent semantic analysis, clustering and Wordnet used in this work; it is presented in Section IV the proposed method detailing its operation and experiments, and in section V the conclusions of the article.

II. Related Work

There are many works in the literature that deal with generation or extraction of ontologies. Most of the works focus on certain documents types or on specific domains.

Initially, it is presented solutions related to ontologies generation using algorithms such as clustering and latent semantic that are relatively independent of the document type, since they only use the textual content of the documents for the ontologies creation.

The work of Maddi et al. [2] presents a way to extract ontologies for text documents using singular value decomposition (SVD) to obtain the concepts from terms and represents the obtained results using bipartite graphs.

Fortuna et al. [3] present a process for obtaining concepts semi-automatically, because the solution only suggests terms sets and from this suggestion the user chooses the concepts and makes connections between them.

Still considering the use of latent semantic, Yeh and Yang [4] generate ontologies from historical documents from digital libraries, using latent semantics for generating the initial concepts and clustering for the other concepts. Regarding the semantic relation generation, the paper proposes the use of a specific set of pre-defined relations to the language and document domain.

Some paper presents detailed studies on the generation of concepts and ontologies. Thus, the state of art for methods, techniques and tools to the ontologies generation is presented in [5, 6, 7], and in [3] it is presents a study of concepts generation focused on clustering and latent semantic.

Considering the solutions that generate ontologies for applications and specific document types, there is the work of Sanchez and Moreno [8] that presents a methodology for automatic construction of domain ontologies in which concepts are obtained of keywords from Web pages. The ontologies creation for lecture notes in distance education systems is presented in [9] and it uses natural language processing to extract keywords, algorithms based on frequency to select the concepts from the keywords and association rules algorithms to define the semantic relations.

Gillam and Ahmad [10] propose the obtainment of concepts using statistical methods for comparison between a vocabulary created by domain experts and the general vocabulary words from the text. For the hierarchy creation it is used solutions from literature, such as smoothing and extraction and placement technique.

Lee et al. [11] present a solution for creating ontologies from text document in Chinese using fuzzy logic, similarity and clustering to obtain the taxonomy of the ontology.

The works presented in the literature are generally directed to a particular area or document type, whereas the proposed method is developed to meet different domains and document types.

Most solutions in the literature generate, as an answer, an ontology that can be manipulated by only using the tool that develops the solution, limiting the use of ontology developed or requiring an adaptation for use in other environments. Thus, the proposed method generates a standard OWL ontology that can be accessed and manipulated in ontology editors or other tools, for example, Gena when programmed in Java.

Another consideration that must be made about the solutions for creating ontologies is that solutions from the literature require the intervention of a specialist to obtain the semantic relations or algorithm that take much time and effort. Therefore, the proposed method uses a simple and relatively quick way to automatically generate the basic semantic relations between the concepts, generating an ontology that has the properties, axioms and constraints on its outcome.

III. CONCEPTS

In this section, it is presented some concepts and techniques used in the development of the proposed method.

A. Latent Semantic Analysis and Singular Value Decomposition (SVD)

Latent Semantic Analysis is a way to manipulate sets of documents [12]. However, in the context of this work, it is used to obtain concepts that comprise a set of documents [2].

The latent semantic analysis explores the relation between terms and documents to build a vector space, which allows the performing of analyzes between documents. To apply the latent semantic index-terms must be obtained which are the most frequent terms in the documents. From the index terms, it is mounted a term-document matrix containing the terms in rows and the term frequency in columns for each of the documents. As the document-term matrix can be very large to be fully analyzed, the SVD is used to obtain an approximation of this matrix through linear combinations.

The SVD decomposes the term-document matrix into three matrices U, Σ and V, where U is an orthonormal matrix whose columns are called singular vectors to the left, Σ is a diagonal matrix whose elements are called not negative singular values and V is an orthonormal matrix whose columns are called singular vectors to the right. Fig. 1 shows the decomposition of an A document-term matrix with dimensions mxn, resulting in matrices U with dimensions mxr, Σ with dimensions rxr and V with dimensions rxn.

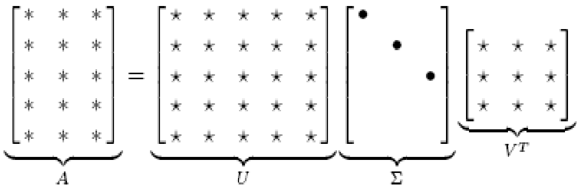


Figure 1. Example of singular value decomposition for a term-document matrix A.

The use of SVD allows both dimensionality reduction of term-document matrix for an information recovery task and the creation of concepts and their association with the document.

The creation of concepts is performed by analyzing the two matrices term-document and U. The first level (ground level) of the ontology hierarchy is obtained from its own index terms from the term-document matrix. The next level of the hierarchy is formed of concepts obtained from the term analysis of the matrix U columns, which provides the relation between terms and concepts. A concept consists of chosen terms from each column according to some criterion.

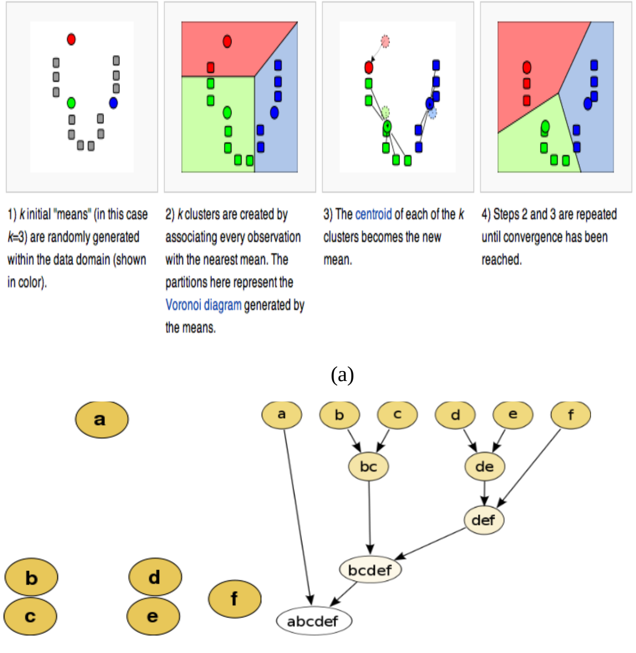
The matrix V provides the relation between concepts obtained from the U matrix and the documents of collection, allowing one to know which concepts are from each document and create a descriptive ontology for each document.

B. Hierarchical Clustering Algorithms

The clustering algorithms in the context of this method are used to perform an analysis on concepts obtained to generate the other levels of the hierarchy of the ontologies. Thus, this section presents the concepts related to hierarchical clustering.

There are two ways to implement hierarchical clustering: bottom-up and top-down. The bottom-up solution starts with several individual concepts that are grouped together with more similar ones until it forms a single group. On the other hand, a top-down solution starts with all objects in one group and these are subdivided according to their proximity in smaller groups.

Among the various bottom-up clustering algorithms, there are two that are most commonly used for creating ontology hierarchies. The K-Means algorithm was presented in 1967 and it begins at the choice of baseline groups (centroids). The algorithm works by arranging objects according to these centroids and recalculating these centers until the result of convergence is satisfactory [13]. However, the clustering algorithm initially considers that all objects are separate groups. The algorithm analyzes the similarity between the two groups putting them together based on the proximity between the groups until there is only one group. Fig. 2 illustrates the operation of K-Means clustering algorithms and clustering.



1) *k* initial "means" (in this case *k*=3) are randomly generated within the data domain (shown in color).

2) *k* clusters are created by associating every observation with the nearest mean. The partitions here represent the Voronoi diagram generated by the means.

3) The centroid of each of the *k* clusters becomes the new mean.

4) Steps 2 and 3 are repeated until convergence has been reached.

(a)

(b)

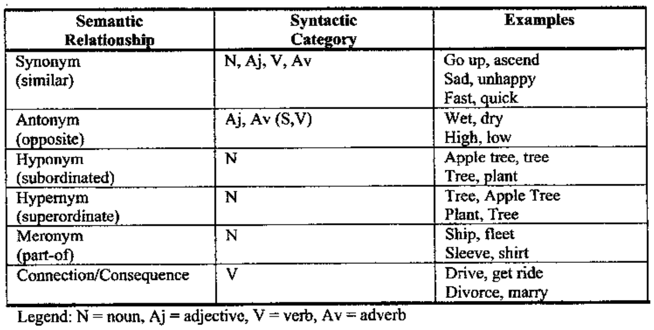
Figure 2. Example of operation of clustering algorithms that can be used in the building of ontologies [13].

C. Wordnet Ontology

Wordnet [14] may be considered an ontology constructed in a more general way or also a lexical reference which can be used online or locally. According to Snasel et al. [15], Wordnet has information of nouns, adjectives, verbs and adverbs, which can be used to determine semantic connections and to trace the connections between morphological words.

Generally, there is a version of Wordnet for each language. However, there are tools to extend the analysis in one language to others, for example, if the noun "house" is analyzed to obtain synonyms, using the tool, all its synonyms may be obtained for English or for any other language.

In this method context, Wordnet is used to create the semantic relations between the ontology concepts focusing on the creation of properties, axioms and restrictions. For the creation of these relations are analyzed possible relations proposed in Wordnet, as shown in Fig. 3.



Semantic Relationship	Syntactic Category	Examples
Synonym (similar)	N, Aj, V, Av	Go up, ascend Sad, unhappy Fast, quick
Antonym (opposite)	Aj, Av (S,V)	Wet, dry High, low
Hyponym (subordinated)	N	Apple tree, tree Tree, plant
Hypernym (superordinate)	N	Tree, Apple Tree Plant, Tree
Meronym (part-of)	N	Ship, fleet Sleeve, shirt
Connection/Consequence	V	Drive, get ride Divorce, marry

Legend: N = noun, Aj = adjective, V = verb, Av = adverb

Figure 3. Wordnet Semantic Relationships [16].

IV. Proposed Method

The proposed method presents a simple, rapid and automatic way of obtaining an initial organization of concepts from collection of any documents that can be formed only by text or structure and text. This proposal aims to meet applications that require semantic descriptions that are meaningful only enough to meet the application and does not need much detail. This method improves some solutions that make use of clustering and statistical methods in order to obtain more significant ontologies by improving the development of concepts and semantic relations. In this method it is possible to obtain an ontology that describes the concepts of an individual document or of a collection of documents. The method seeks to work only on an automated way, making a specialist unnecessary at the time of the ontology creation. However, a domain expert may do an analysis of the ontology using an ontology editor and make changes to improve the result obtained automatically. The method also keeps stored summaries and elements used to obtain the concepts and terms, so that this method allows the inclusion of new documents in the collection, as well as the deletion and alteration.

Fig. 4 shows the method general outline of ontologies extraction from a collection of documents. In the following sections the main parts of the method and the results obtained using it are shown.

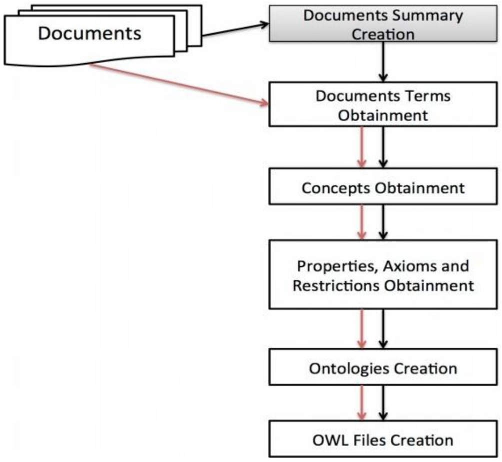

Documents
Documents Summary Creation
Documents Terms Obtainment
Concepts Obtainment
Properties, Axioms and Restrictions Obtainment
Ontologies Creation
OWL Files Creation

Figure 4.General outline of the proposed method operation.

A. Documents Preparation

In this first phase, documents are prepared for obtaining concepts. First, it is necessary that the collection of documents be analyzed in order to define which documents have structure (XML documents) or text only.

Considering only text documents, initially, it is analyzed the necessity of obtaining a summary if the document is very large. For summaries preparation can be used one of the algorithms present in the literature depending on the desired quality and efficiency. The summaries of documents can be kept stored in files or databases, since their preparation need a reasonable computational time that can be suppressed by keeping them for use in the preparation of other ontologies when these documents are used.

For documents that have structure, there is a prior step to the summaries creation. This step is the separation between structure and content of the document. In this separation, the structure is analyzed to verify if the elements have definitions that can be considered concepts in the ontology. The elements are ignored if the structure does not have relevant ones, otherwise they are also stored. The separated contents are analyzed following the same idea of only text documents.

The summaries / documents are read, extracting the terms that will be used in the preparation of the ontology, i.e., these are transformed into set of strings containing terms not repeated and considered relevant of each of the summaries / documents.

These terms also undergo a standardization process, that is, the terms are analyzed in order to withdraw from the set terms that are grammatically different forms for the same word, such as student and students, and terms that are different tenses for the same verb, for example, walk and walks. For XML documents, the term set can contain structure elements that are relevant to the formation of concepts.

Still at this stage, the terms need to have their TF-IDF (Term Frequency Inverse Document Frequency) calculated. The TF-IDF is calculated in two steps, first TF is obtained by the formula presented in (1):

$$TF= freq\_(i,j)/max(freq\_(l,i)) \quad (1)$$

where freq_(i,f) is the frequency of term i for a document j and e max(freq_(l,j)) is the frequency of the most frequent term in the document. However, the IDF is the second stage of the calculation, and it is obtained by the formula (2) shown below:

$$IDF= log 〖N/n\_i 〗 \quad (2)$$

where N is the total number of documents of the set and n_i is the number of documents that contain term i. The final result of TF-IDF is obtained by multiplying the TF by the IDF. The TF-IDF is used in the next phase, in getting the concepts.

B. Concepts Obtainment

Initially, it is obtained the index terms, which are the set of terms that appear in more than twenty five percent of the documents. If this obtained set of terms is very large, it can be reduced by selecting a subset of these terms observing the criterion of keeping in the index the terms that appear more frequently in the documents, so that the manipulation of the document-term matrix and of the matrices created by SVD become easier.

For the resulting matrices from the application of SVD in term-document matrix, the matrices U that links the concepts to the terms and V matrix that links the concepts to documents are used.

The use of the matrix U in order to obtain the ontology concepts has been shown in [2, 3]. The concepts are created from the terms of the matrix U columns. Thus, each column from U creates a concept from the union of the terms that have the highest values in the column, with maximum of three terms united. A comparative analysis is made in this obtained set of concepts in order to verify the concepts that may be the same, i.e., those having the same terms only placed differently. If the concepts are actually different, they are kept in set of concepts and the terms are attached to these concepts. The linking between the terms and concepts is done through analysis of matrix U, verifying in each column the terms that have values greater than 0.5, because the relation is only considered valid if the connectivity degree is greater than fifty percent.

The obtained concepts from the matrix U are the ones of the intermediate level of the ontology, that is, the second level in the hierarchy. At the base level of the ontology, there are the initial concepts that are themselves index terms.

Fig. 5 shows an obtained concept in one experiment performed and its terms, with concept being formed of two terms with the greatest value in the matrix U column.

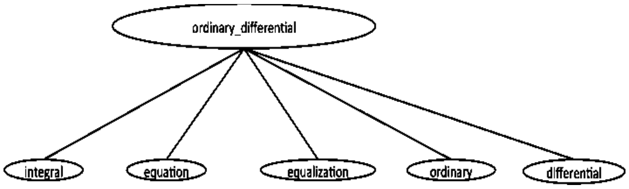
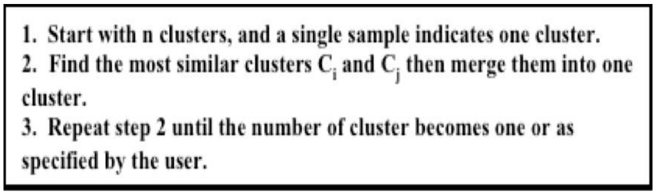


Figure 5.Example of a concept and its terms.

From the two obtained levels of concepts, it is necessary to create the other levels of the ontology to form a complete hierarchy. Thus, it is used the algorithm shown in Fig. 6 for clustering the concepts until obtaining an only group which will be the main concept of hierarchy.



1. Start with n clusters, and a single sample indicates one cluster.
2. Find the most similar clusters C_i and C_j then merge them into one cluster.
3. Repeat step 2 until the number of cluster becomes one or as specified by the user.

Figure 6. Agglomerative Algorithm.

At this point you need to check which concepts belong to which documents, because only the concepts of each one of the documents are clustered. Thus, before clustering, it is necessary that the matrix V be analyzed by separating the concepts and terms of each of the documents. As the terms, the concepts also have its degree of connectivity analyzed, and it is considered for the document only those concepts that have connectivity superior to fifty percent, therefore ensuring greater quality in the developed ontology.

C. Creation of properties, Axioms and Restrictions

After defining the concepts of each document, it is obtained the semantic relations for each one of the ontologies. These relations are organized into properties, axioms and constraints. Two types of properties can be defined: the object properties and data type properties. Object properties relate instances with other instances defining restrictions and behaviors. Data types refer to properties that express only values, e.g., strings or numbers. The concepts can have super and sub-concepts, providing a rationalizing mechanism and property inheritance. Finally, the axioms are used to provide information about the concepts and properties, such as, to specify the equivalence of two concepts or range of a property.

There are many semantic relations that can be obtained using Wordnet. Initially, it is set up the simplest of the properties, which is the subclass_of between concepts of different levels that form the ontology. After, other relations like, equivalent_to (between synonyms or similar concepts), disjoin_of (between antonyms), part_of (between terms that complete others) and inverse_of (between antonyms and synonyms), can be defined. To define these relations, the concepts are analyzed using Wordnet, verifying possible correlations between the considered concepts. For these found correlations, it is analyzed the ones which are suitable for the use in the ontology definition, for example, if the concepts are synonyms, they are given an equivalence defined axiom. In this work, only Wordnet ontology was used to obtaining these correlations, however, depending on the document field, other ontologies may be used.

Besides Wordnet, the concepts are also analyzed for their degree of similarity. Depending on this similarity value, the concepts receive the semantic relation of equivalence. For this work, it was accepted as equivalent the concepts which have a degree of similarity greater than 0.90.

To simplify the process of the semantic relations obtainment, the analysis is performed by level, i.e., the concepts of a same level are examined in pairs until all possible relations are defined.

Fig. 7 presents the semantic relations defined for the concept in Fig. 5, being these relations are: subclass_of between the concept and the terms, and equivalant_to for synonymous terms.

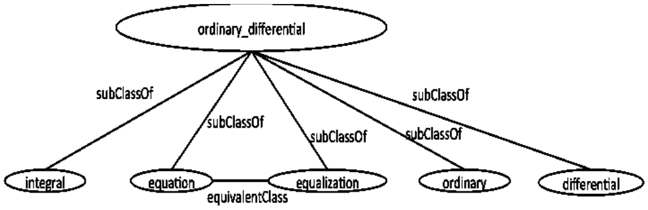


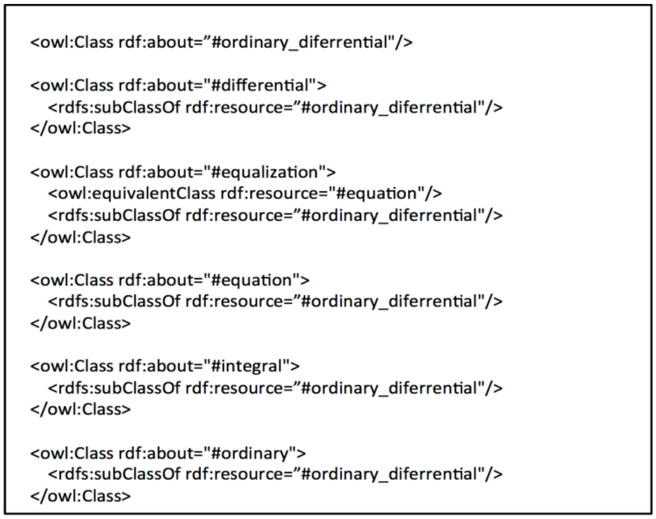
Figure 7. Example of defined properties for a concept.

D. Ontology Creation

This is last phase of method. The concept and semantic relations are organized in ontologies that are stored in files encoded in OWL language. This language is used to define ontologies and it provides mechanisms for component creation: concepts, instances, properties and axioms.

As a result of this phase, there is a set of ontologies in OWL for the documents in the collection. However, using all concepts and relations obtained, a single ontology describing the entire collection can be created, thus creating an ontology that may be worked by a specialist to form an ontology of domain.

Fig. 8 shows an example of a possible OWL coding to the concepts of Fig. 7.

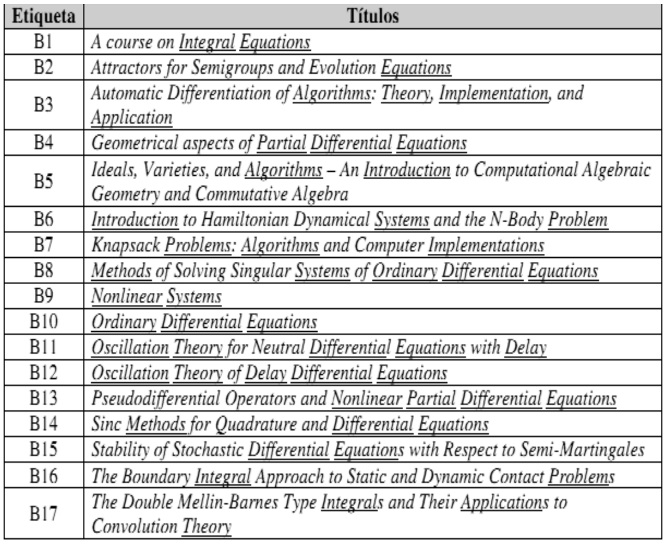


```
<owl:Class rdf:about=”#ordinary_diferrential"/>

<owl:Class rdf:about="#differential">
  <rdfs:subClassOf rdf:resource=”#ordinary_diferrential"/>
</owl:Class>

<owl:Class rdf:about="#equalization">
  <owl:equivalentClass rdf:resource="#equation"/>
  <rdfs:subClassOf rdf:resource=”#ordinary_diferrential"/>
</owl:Class>

<owl:Class rdf:about="#equation">
  <rdfs:subClassOf rdf:resource=”#ordinary_diferrential"/>
</owl:Class>

<owl:Class rdf:about="#integral">
  <rdfs:subClassOf rdf:resource=”#ordinary_diferrential"/>
</owl:Class>

<owl:Class rdf:about="#ordinary">
  <rdfs:subClassOf rdf:resource=”#ordinary_diferrential"/>
</owl:Class>
```

Figure 8. OWL codification to the concepts of Fig. 7.

E. Experimental Results

To validate the proposed method, ontologies were created for both text and XML collections of documents. The descriptions of collections and of obtained results are provided below. The first experiment creates ontologies for a simple collection of documents with small texts about book titles. The group has seventeen documents, as shown in Fig. 9.

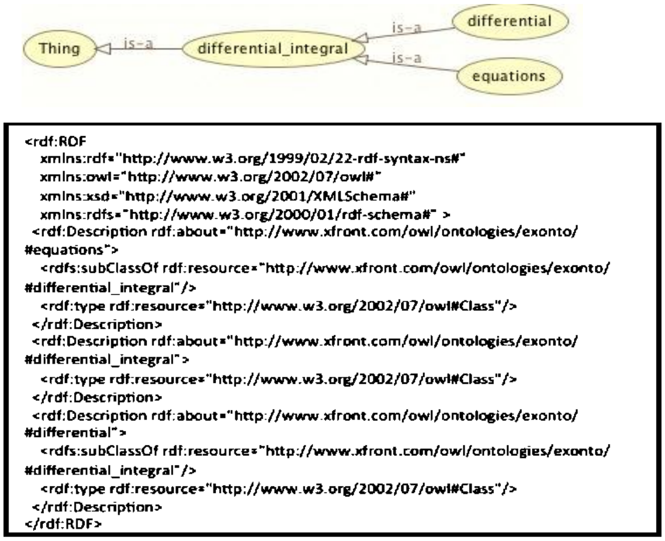


Etiqueta	Títulos
B1	*A course on Integral Equations*
B2	*Attractors for Semigroups and Evolution Equations*
B3	*Automatic Differentiation of Algorithms: Theory, Implementation, and Application*
B4	*Geometrical aspects of Partial Differential Equations*
B5	*Ideals, Varieties, and Algorithms – An Introduction to Computational Algebraic Geometry and Commutative Algebra*
B6	*Introduction to Hamiltonian Dynamical Systems and the N-Body Problem*
B7	*Knapsack Problems: Algorithms and Computer Implementations*
B8	*Methods of Solving Singular Systems of Ordinary Differential Equations*
B9	*Nonlinear Systems*
B10	*Ordinary Differential Equations*
B11	*Oscillation Theory for Neutral Differential Equations with Delay*
B12	*Oscillation Theory of Delay Differential Equations*
B13	*Pseudodifferential Operators and Nonlinear Partial Differential Equations*
B14	*Sinc Methods for Quadrature and Differential Equations*
B15	*Stability of Stochastic Differential Equations with Respect to Semi-Martingales*
B16	*The Boundary Integral Approach to Static and Dynamic Contact Problems*
B17	*The Double Mellin-Barnes Type Integrals and Their Applications to Convolution Theory*

Figure 9. Presentation of the experiment documents and their contents [12].

As it is a collection with very short texts there is no need to create summaries. Thus, the method begins obtaining the terms sets of documents, on which it is applied the latent semantic technique and the other method steps for the building of ontologies.

Fig. 10 shows encoding OWL of ontology of document B4 and its graph generated in the Protégé editor available at [17]. Fig. 11 shows created ontology of document B11 where it can be seen a larger number of semantic relations between elements.



```
<rdf:RDF
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:owl="http://www.w3.org/2002/07/owl#"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema#"
  xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#" >
 <rdf:Description rdf:about="http://www.xfront.com/owl/ontologies/exonto/
#equations">
  <rdfs:subClassOf rdf:resource="http://www.xfront.com/owl/ontologies/exonto/
#differential_integral"/>
  <rdf:type rdf:resource="http://www.w3.org/2002/07/owl#Class"/>
 </rdf:Description>
 <rdf:Description rdf:about="http://www.xfront.com/owl/ontologies/exonto/
#differential_integral">
  <rdf:type rdf:resource="http://www.w3.org/2002/07/owl#Class"/>
 </rdf:Description>
 <rdf:Description rdf:about="http://www.xfront.com/owl/ontologies/exonto/
#differential">
  <rdfs:subClassOf rdf:resource="http://www.xfront.com/owl/ontologies/exonto/
#differential_integral"/>
  <rdf:type rdf:resource="http://www.w3.org/2002/07/owl#Class"/>
 </rdf:Description>
</rdf:RDF>
```

Figure 10. Example of generated ontology for document B4 of the collection presented in Fig. 9.

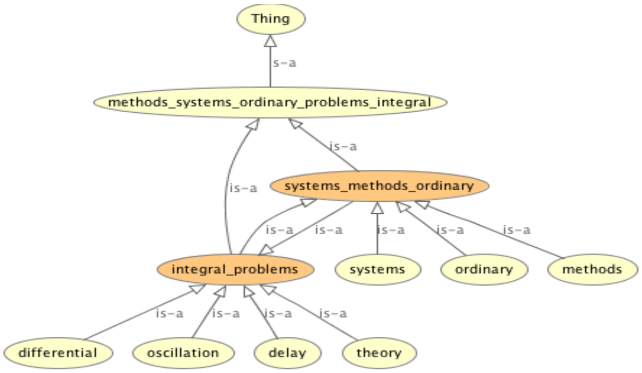
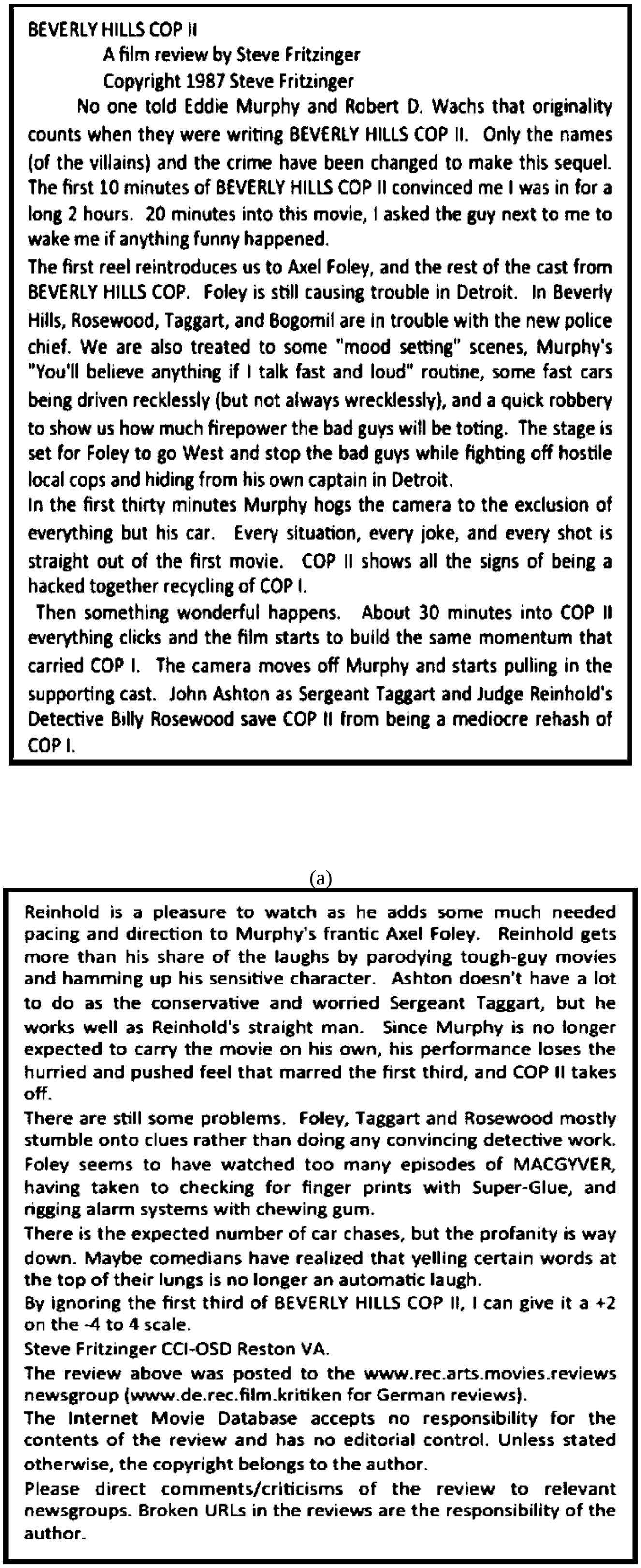


Figure 11. Generated ontology for document B11 of the collection presented in Fig. 9.

Still considering only text documents, a second experiment was carried out using documents with larger size, which require the preparation of a summary. The collection has fifteen documents chosen randomly from a collection of twenty-seven thousand documents about international movie reviews. From this collection, the ontologies for each one of the documents and for the whole collection of documents were obtained. Fig. 12 shows in (a), a text of document example; in (b), terms of the generated summary for the document; and in (c), the ontology created for the document.



BEVERLY HILLS COP II

A film review by Steve Fritzinger

Copyright 1987 Steve Fritzinger

No one told Eddie Murphy and Robert D. Wachs that originality counts when they were writing BEVERLY HILLS COP II. Only the names (of the villains) and the crime have been changed to make this sequel. The first 10 minutes of BEVERLY HILLS COP II convinced me I was in for a long 2 hours. 20 minutes into this movie, I asked the guy next to me to wake me if anything funny happened.

The first reel reintroduces us to Axel Foley, and the rest of the cast from BEVERLY HILLS COP. Foley is still causing trouble in Detroit. In Beverly Hills, Rosewood, Taggart, and Bogomil are in trouble with the new police chief. We are also treated to some "mood setting" scenes, Murphy's "You'll believe anything if I talk fast and loud" routine, some fast cars being driven recklessly (but not always wrecklessly), and a quick robbery to show us how much firepower the bad guys will be toting. The stage is set for Foley to go West and stop the bad guys while fighting off hostile local cops and hiding from his own captain in Detroit.

In the first thirty minutes Murphy hogs the camera to the exclusion of everything but his car. Every situation, every joke, and every shot is straight out of the first movie. COP II shows all the signs of being a hacked together recycling of COP I.

Then something wonderful happens. About 30 minutes into COP II everything clicks and the film starts to build the same momentum that carried COP I. The camera moves off Murphy and starts pulling in the supporting cast. John Ashton as Sergeant Taggart and Judge Reinhold's Detective Billy Rosewood save COP II from being a mediocre rehash of COP I.

(a)

Reinhold is a pleasure to watch as he adds some much needed pacing and direction to Murphy's frantic Axel Foley. Reinhold gets more than his share of the laughs by parodying tough-guy movies and hamming up his sensitive character. Ashton doesn't have a lot to do as the conservative and worried Sergeant Taggart, but he works well as Reinhold's straight man. Since Murphy is no longer expected to carry the movie on his own, his performance loses the hurried and pushed feel that marred the first third, and COP II takes off.

There are still some problems. Foley, Taggart and Rosewood mostly stumble onto clues rather than doing any convincing detective work. Foley seems to have watched too many episodes of MACGYVER, having taken to checking for finger prints with Super-Glue, and rigging alarm systems with chewing gum.

There is the expected number of car chases, but the profanity is way down. Maybe comedians have realized that yelling certain words at the top of their lungs is no longer an automatic laugh.

By ignoring the first third of BEVERLY HILLS COP II, I can give it a +2 on the -4 to 4 scale.

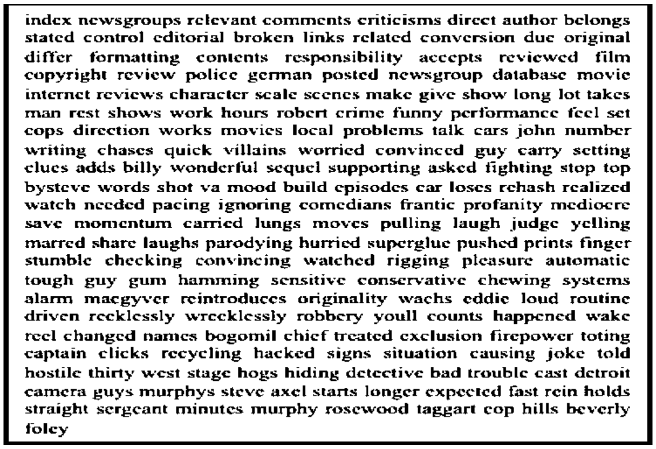
Steve Fritzinger CCI-OSD Reston VA.

The review above was posted to the www.rec.arts.movies.reviews newsgroup (www.de.rec.film.kritiken for German reviews).

The Internet Movie Database accepts no responsibility for the contents of the review and has no editorial control. Unless stated otherwise, the copyright belongs to the author.

Please direct comments/criticisms of the review to relevant newsgroups. Broken URLs in the reviews are the responsibility of the author.

(b)



index newsgroups relevant comments criticisms direct author belongs stated control editorial broken links related conversion due original differ formatting contents responsibility accepts reviewed film copyright review police german posted newsgroup database movie internet reviews character scale scenes make give show long lot takes man rest shows work hours robert crime funny performance feel set cops direction works movies local problems talk cars john number writing chases quick villains worried convinced guy carry setting clues adds billy wonderful sequel supporting asked fighting stop top bysteve words shot va mood build episodes car loses rehash realized watch needed pacing ignoring comedians frantic profanity mediocre save momentum carried lungs moves pulling laugh judge yelling marred share laughs parodying hurried superglue pushed prints finger stumble checking convincing watched rigging pleasure automatic tough guy gum hamming sensitive conservative chewing systems alarm macgyver reintroduces originality wachs eddie loud routine driven recklessly wrecklessly robbery youll counts happened wake reel changed names bogomil chief treated exclusion firepower toting captain clicks recycling hacked signs situation causing joke told hostile thirty west stage hogs hiding detective bad trouble cast detroit camera guys murphys steve axel starts longer expected fast rein holds straight sergeant minutes murphy rosewood taggart cop hills beverly foley

(c)

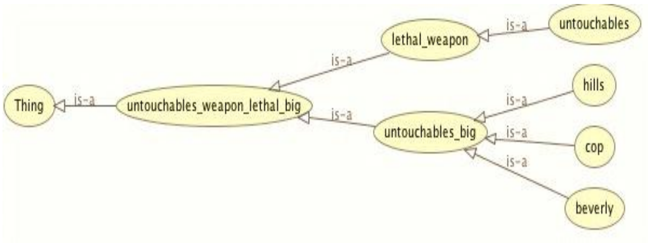


Figure 12. Text example, summary and ontology created. In part (a), the document was presented; in part (b), the summary generated for the document; and in part (c), the ontology for this document.

As presented in the previous sections, it is possible to generate an ontology that describes the concepts and terms of the whole collection of documents. Thus, Fig. 13 shows the generated ontology for the experiment fifteen documents.

The third experiment was carried out with a collection of twenty-four XML documents about historical manuscripts. For the development of this experiment, first it was carried out the separation between structure and content.

After the separation between the documents structure and content, the analysis was performed to verify if the structure had relevant information for the ontology formation. In the case of used documents in experiment, the structure is nothing but the structuring of text sections, so only the content was used.

Since each document in the set has a considerable number of pages, the number of terms in summaries is high, and also the index-terms set, complicating the matrix manipulation at the concept obtainment time. Thus, it was considered for this experiment only the five hundred more frequent terms in the documents to obtain the index-terms. Applying the proposed method, ontologies have been created for each of the documents and for the collection.

Fig. 14 shows the ontology created to the collection of document, demonstrating concepts, terms and semantic relations

The carried out experiments showed that the individual ontologies generated to documents express significantly, even though simply, the concepts contained therein. For example, for the document shown in Figure 12, it is possible to notice that the film described in the presented review has to do with a local (Beverly Hill), police and violence (lethal and weapon). As for the document B4, it is possible to know that the document is a book about some aspect of differential equations as show the ontology concepts, differential and equations.

These experiments demonstrated that the method satisfactorily obtained the semantic relations between concepts and terms simply and automatically, improving the created ontology, because it can be identified similar terms by synonymy and other relations due to the use of similarity and Wordnet. In order to improve the obtained semantic relations in the created ontologies, it would be possible to use other ontologies or a thesaurus besides Wordnet.

The proposed method has fulfilled its proposition because even though it is very simple, the use of Wordnet combined with the employed techniques have improved the obtained results, allowing better definition of document concepts and the semantic relations that compose the generated ontology.

The resulting ontologies are stored in an OWL file that can be edited or viewed by the usual ontology editors, allowing its easier handling.

V Conclusion

This paper presented a method for document or collection of documents ontology extraction using latent semantic, clustering and Wordnet. The proposed method is fully automatic and simple, but with significant results enough to allow the understanding and manipulation of the document concepts without needing advanced techniques, the intervention of an expert, or even the entire understanding of the domain.

The experiments showed that the obtained ontologies satisfactorily represent the concepts of the documents. Despite that, this method can still be improved using other tools and techniques that allow the definition of other semantic relations between the concepts and enhance the concepts obtainment.

References

[1] K. Breitman, Web Semantica - A internet do Futuro, vol. 1, Rio de Janeiro, RJ: LTC, 2006, p. 190.

[2] G. R. Maddi, C. S. Velvadapu, S. Strivastava e J. G. d. Lamadrid, “Ontology Extraction from Text Documents by Singular Value Decomposition,” em ADMI 2001, 2001.

[3] B. Fortuna, D. Mladenic e M. Grobelniz, “Semi-automatic Construction of Topic Ontology,” em Lecture Notes in Computer Science, vol. 4289, Springer, 2005, pp. 121-131.

[4] J. Yeh e N. Yang, “Ontology Construction on Latent Topic Extraction in a Digital Library,” em International Conference on Asian Digital Libraries 2008, 2008.

[5] Y. Ding e S. Foo, “Ontology Research and Development Part 1 – A Review of Ontology Generation,” Journal of Information Science, vol. 28, pp. 123-136, 2002.

[6] I. Bedini e B. Nguyen, “Automatic ontology generation: State of the art,” 2007.

[7] A. Zauaq, “A survey of Domain Engineering: Method and Tools,” em Studies in Computational Intelligence, vol. 308, 2010, pp. 103-119.

[8] D. Sanchez e A. Moreno, “Creating ontologies from Web documents,” Recent Advances in Artificial Intelligence Research and Development, vol. 113, pp. 11-18, 2004.

[9] N. Gantayat, “Automated Construction of Domain Ontologies from Lecture Notes,” Bombay, 2011.

[10] L. Gillam e K. Ahmad, "Automatic Ontology Extraction from Unstructured Texts," em the Move to Meaningful Internet Systems 2005: CoopIS, DOA, and ODBASE, 2005.

[11] C.S. Lee, Y.-F. Kao, Y.-H. Kou e M.-H. Wang, "Automated Ontology Construction for Unstructured Text Documents," Data & Knowledge Engineering, 2007.

[12] D. Foronda, "Estudo Exploratório da Indexação Semantica Latente," PUC-RS, Porto Alegre.

[13] [13] B. P. Nunes, "Classificação automatica de dados semi-estruturados," Rio de Janeiro, 2009.

[14] "Wordnet," [Online]. Available: http://wordnet.princeton.edu. [Accessed in 2012].

[15] V. Snasel, P. Maravec e J. Pokorney, "Wordnet Ontology based Model for Web Retirval," em WIRI '05 Proceedings of the International Workshop on Challenges in Web Information Retrieval and Integration, 2005.

[16] K. Breitman, M. A. Casanova e W. Truszkowski, Semantic Web: Concepts, Technologies and Applications, Springer, 2007.

[17] "Protégé" [Online]. Available: http://protege.stanford.edu. [Accessed in 2012]

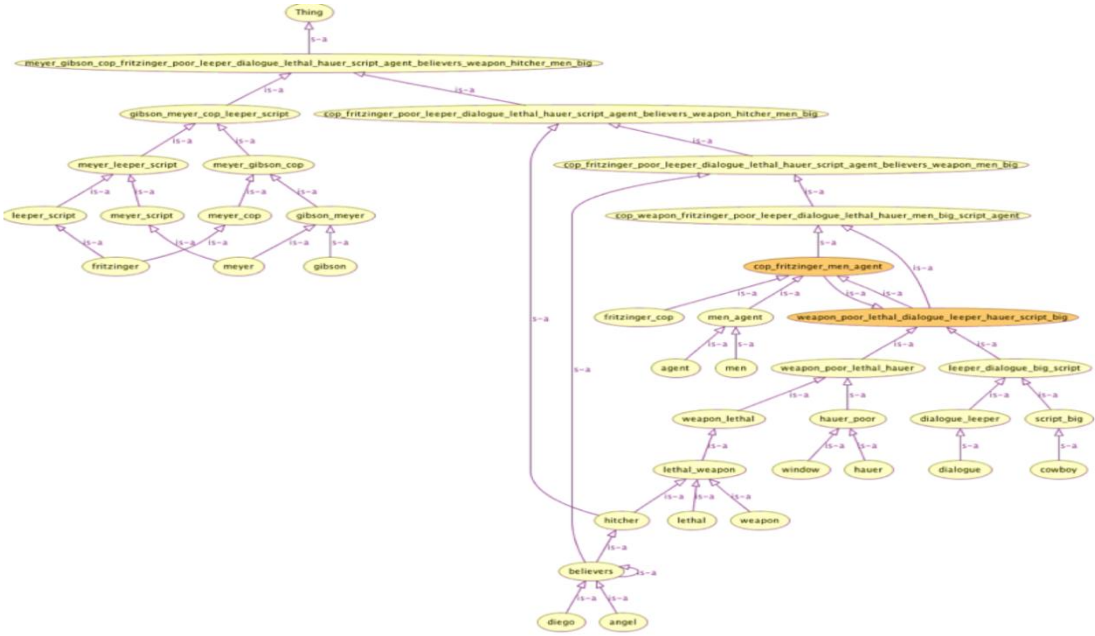

Thing
is-a
meyer_gibson_cop_fritzinger_poor_leeper_dialogue_lethal_hauer_script_agent_believers_weapon_hitcher_men_big
gibson_meyer_cop_leeper_script
cop_fritzinger_poor_leeper_dialogue_lethal_hauer_script_agent_believers_weapon_hitcher_men_big
meyer_leeper_script
meyer_gibson_cop
cop_fritzinger_poor_leeper_dialogue_lethal_hauer_script_agent_believers_weapon_men_big
leeper_script
meyer_script
meyer_cop
gibson_meyer
cop_weapon_fritzinger_poor_leeper_dialogue_lethal_hauer_men_big_script_agent
fritzinger
meyer
gibson
cop_fritzinger_men_agent
fritzinger_cop
men_agent
weapon_poor_lethal_dialogue_leeper_hauer_script_big
agent
men
weapon_poor_lethal_hauer
leeper_dialogue_big_script
weapon_lethal
hauer_poor
dialogue_leeper
script_big
lethal_weapon
window
hauer
dialogue
cowboy
hitcher
lethal
weapon
believers
diego
angel

Figure 13. Generated ontology for the collection of fifteen text-only documents.

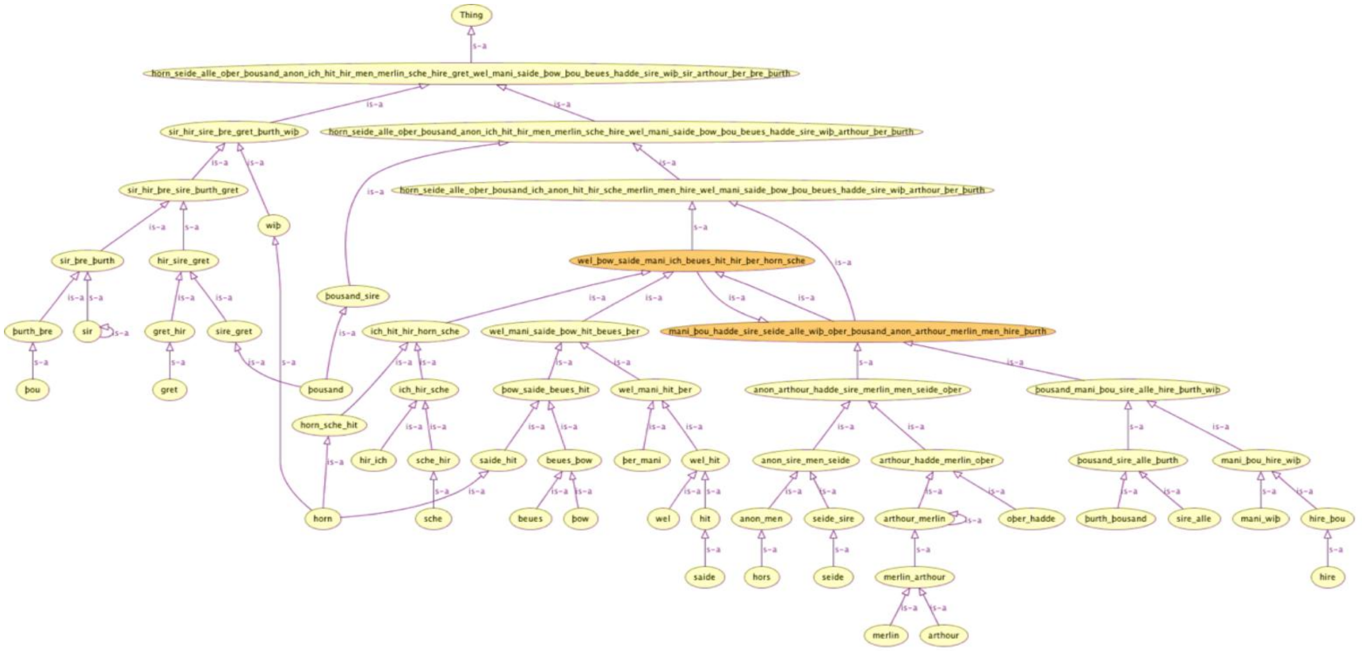

Thing
is-a
horn_seide_alle_oþer_þousand_anon_ich_hit_hir_men_merlin_sche_hire_gret_wel_mani_saide_þow_þou_beues_hadde_sire_wiþ_sir_arthour_þer_þre_þurth
sir_hir_sire_þre_gret_þurth_wiþ
horn_seide_alle_oþer_þousand_anon_ich_hit_hir_men_merlin_sche_hire_wel_mani_saide_þow_þou_beues_hadde_sire_wiþ_arthour_þer_þurth
sir_hir_þre_sire_þurth_gret
wiþ
horn_seide_alle_oþer_þousand_ich_anon_hit_hir_sche_merlin_men_hire_wel_mani_saide_þow_þou_beues_hadde_sire_wiþ_arthour_þer_þurth
sir_þre_þurth
hir_sire_gret
wel_þow_saide_mani_ich_beues_hit_hir_þer_horn_sche
þousand_sire
þurth_þre
sir
gret_hir
sire_gret
ich_hit_hir_horn_sche
wel_mani_saide_þow_hit_beues_þer
mani_þou_hadde_sire_seide_alle_wiþ_oþer_þousand_anon_arthour_merlin_men_hire_þurth
þou
gret
þousand
ich_hir_sche
þow_saide_beues_hit
wel_mani_hit_þer
anon_arthour_hadde_sire_merlin_men_seide_oþer
þousand_mani_þou_sire_alle_hire_þurth_wiþ
horn_sche_hit
hir_ich
sche_hir
saide_hit
beues_þow
þer_mani
wel_hit
anon_sire_men_seide
arthour_hadde_merlin_oþer
þousand_sire_alle_þurth
mani_þou_hire_wiþ
horn
sche
beues
þow
wel
hit
anon_men
seide_sire
arthour_merlin
oþer_hadde
þurth_þousand
sire_alle
mani_wiþ
hire_þou
saide
hors
seide
merlin_arthour
hire
merlin
arthour

Figure14. The ontology representing the whole experiment collection of XML documents.