

2013 10th International Conference on Information Technology: New Generations

Analysing the Problem and Main Approaches for Ontology Population

Carla Faria, Rosario Girardi
Computer Science Department
Federal University of Maranhão
São Luiz, Brazil

carlafaria@ifma.edu.br, rosariogirardi@gmail.com

Paulo Novais
Computer Science Department
University of Minho
Braga, Portugal
pjon@di.uminho.pt

Abstract— Knowledge systems are a suitable computational approach to solve complex problems and to provide decision support. Ontologies are an approach for knowledge representation and Ontology Population looks for instantiating the constituent elements of an ontology, like properties and non-taxonomic relationships. Manual population by domain experts and knowledge engineers is an expensive and time consuming task. Thus, automatic or semi-automatic approaches are needed. This paper discusses the problem of Automatic Ontology Population and proposes a generic process specifying its phases and what kind of techniques can be used to perform the activities of each phase. Some techniques representing the state of the art of this field are also described along with the solutions they adopt for each phase of the AOP process with their advantages and limitations. This work is part of HERMES, a Brazil/Portugal research cooperation project looking for techniques and tools for automating the process of ontology learning and population.

Keywords—ontology; ontology population; information extraction; natural language processing

I. INTRODUCTION

Knowledge systems are a suitable computational approach to solve complex problems and to provide decision support. Their main components are a knowledge base and an inference mechanism to draw conclusions from that knowledge. Knowledge representation formalisms, like ontologies, are used by modern knowledge systems, to represent and share the knowledge of an application domain [20]. Supporting semantic processing, they allow for more precise information interpretation. Thus, knowledge systems can provide greater usability and effectiveness than traditional information systems [12].

Ontology Population (OP) looks for identifying instances of non-taxonomic relationships and properties of ontology classes. Manual population by domain experts and knowledge engineers is an expensive and time-consuming task. Thus, automatic or semi-automatic approaches are needed. Performing a fast and low cost ontology population is crucial for success on the development of knowledge systems.

Most approaches for Automatic Ontology Population (AOP) from texts are based on Natural Language Processing

(NLP) [1], Statistics Information Retrieval (SIR) [21], Machine Learning (ML) [2] and / or Information Extraction (IE) [6]. NLP techniques are used to annotate the corpus with the information needed for subsequent processing. SIR is used to extract from the corpus candidate instances. IE and ML techniques are used to validate the extracted instances and to classify them from the annotated corpus.

This paper discusses the problem of AOP and proposes a generic process specifying its phases and what kind of techniques can be used to perform the activities of each phase. Some techniques of the AOP state of the art are also described along with the solutions they adopt for each phase of the AOP process. This work is part of HERMES, a Brazil/Portugal research cooperation project looking for techniques and tools for automating the process of ontology learning and population.

The article is organized as follows. Section II introduces the formal ontology definition used in this work. Section III proposes a generic process approaching the OP problem along with the techniques that can be applied in each one of its phases. Section IV summarizes current approaches of OP and solutions they adopt according to the process described in section III. Section V describes main techniques for the evaluation of OP techniques and finally, section VI concludes the work.

II. A FORMAL DEFINITION OF AN ONTOLOGY

An ontology is a formal and explicit specification of a shared conceptualization of a domain of interest. Their classes, relationships, constraints and axioms define a common vocabulary to share knowledge [14]. Conceptualization refers to an abstract model of some phenomenon in the world. Explicit, means that the type of concepts used and the limitations of their use are explicitly defined. Formal, refers to the fact that the ontology should be machine readable. Shared, reflects the notion that ontology captures consensual knowledge, that is, it is not private to some individual but accepted by a group.

Formally, an ontology can be defined as the tuple [12]:

$$O = (C, H, I, R, P, A) \quad (1)$$

where,

$C = C^C \cup C^I$ is the set of entities of the ontology. The C^C set consists of classes, i.e., concepts that represent entities (for example, “Person” $\in C^C$) describing a set of objects, class instances in the C^I set (for example “Erik Brown” $\in C^I$).

$H = \{\text{kind_of}(c_1, c_2) \mid c_1 \in C^C, c_2 \in C^C\}$ is the set of taxonomic relationships between concepts, which define a concept hierarchy and are denoted by “kind_of(c_1, c_2)”, meaning that c_1 is a subclass of c_2 , for instance, “kind_of(Lawyer, Person)”.

$I = \{\text{is_a}(c_1, c_2) \mid c_1 \in C^I, c_2 \in C^C\} \cup \{\text{prop}^I(c_k, \text{value}) \mid c_k \in C^I\} \cup \{\text{rel}_k(c_1, c_2, \dots, c_n) \mid \forall i, c_i \in C^I\}$ is the set of instance relationships related to the C^C (eg. “is a (Anne Smith, Client)”), P (eg. “subject (Case₁₂, “adoption”)”) and R (eg. “represents (Erik Brown, Anne Smith)”) sets.

$R = \{\text{rel}_k(c_1, c_2, \dots, c_n) \mid \forall i, c_i \in C^C\}$ is the set of non-taxonomic ontology relationships like “represents(Lawyer, Client)”.

$P = \{\text{prop}^C(c_k, \text{datatype}) \mid c_k \in C^C\}$ is the set of properties of ontology entities. The relationship prop^C defines the basic datatype of a class property. For instance, subject (Case, String) is an example of a prop^C property.

$A = \{\text{condition}_x \Rightarrow \text{conclusion}_y(c_1, c_2, \dots, c_n) \mid \forall j, c_j \in C^C\}$ is a set of axioms, rules that allow checking the consistency of an ontology and infer new knowledge through some inference mechanism. The term condition_x is given by $\text{condition}_x = \{(cond_1, cond_2, \dots, cond_n) \mid \forall z, cond_z \in H \cup I \cup R\}$. For instance, “ $\forall \text{Defense_Argument, OldCase, NewCase, applied_to}(\text{Defense_Argument, OldCase, NewCase})$, similar_to (OldCase, NewCase) \Rightarrow applied_to(Defense_Argument, NewCase)” is a rule that indicates that if two legal cases are similar then, the defense argument used in one case could be applied to the other one.

III. THE PROBLEM OF ONTOLOGY POPULATION

Ontology Population (OP) looks for identifying instances of non-taxonomic relationships and properties of ontology classes. Non-taxonomic relationships and properties instances correspond to the I set of the ontology definition in section II. For example, “represents” is a non-taxonomic relationship between the classes “lawyer” and “client”; “subject” is a property of the class “case” in the domain of a law firm.

Fig. 1 illustrates the generic process we are proposing for OP along with the process phases, their sequence, phase inputs and outputs and supporting techniques. OP is usually accomplished through the three phases: “Identification of Candidate Instances”, “Construction of a Classifier” and “Classification of Instances”. Typically, the “Identification of Candidate Instances” phase occurs through the application of NLP [1] and / or SIR [21] techniques. The “Construction of a Classifier” and “Classification of Instances” phases are performed by applying IE [6] and / or ML [2] techniques. Inputs of the process are a corpus and ontology. The output is a populated ontology. Fig. 2 shows a simple example of

OP illustrating the outputs of each phase of the proposed process in Fig. 1.

The “Identification of Candidate Instances” phase identifies instances of non-taxonomic relationships and properties of ontology classes by annotating the inputted corpus. Usually, this phase uses NLP techniques [Allen 1995], mainly “Morpho-Lexical Analysis”, “Named Entity Recognition” and “Co-References Identification”. “Morpho-Lexical Analysis” aims at identifying the grammatical categories of each token in a sentence. “Named Entity Recognition” identifies names that refer to unique objects in the world such as names of persons, organizations and places. “Co-Reference Identification” identifies both pronoun and nominal co-references. A pronominal co-reference consists of pronouns that refer to previously described entities, while the nominal co-reference consists of nouns that refer to the same entity. Fig. 2 shows an example of an annotated corpus in the legal domain outputted from the phase “Identification of Candidate Instances” with the five detected instances: “Keith R.”, “H. R.”, “Mariana”, “1950” and “2006”.

The “Identification of Candidate Instances” phase uses SIR [21] to identify instances by extracting the terms of a document according to TF-IDF, TF and / or IDF values. The identified terms are those whose weights are within a particular interval. The interval and the statistic measure are experimentally defined. In the experiments conducted, recall and precision were calculated for the extracted instances grouped by TF, IDF and TF-IDF to take those in the interval with greatest number of instances and the best balance between recall and precision. Fig. 2 shows an example of an annotated corpus in the legal domain outputted from the phase “Identification of Candidate Instances” with the three detected instances: “Keith R.”, “H. R.” and “Mariana”

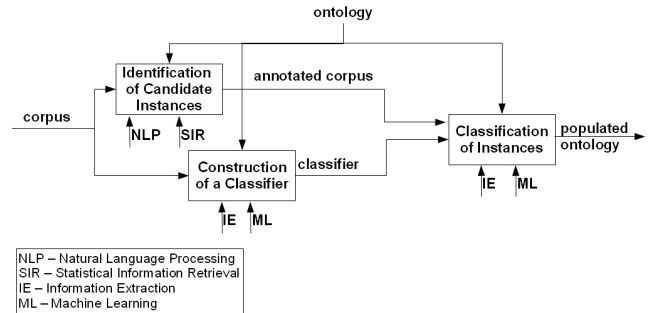


Figure 1. A generic process for automatic ontology population.

The “Construction of a Classifier” phase has a corpus and an ontology as input and generates a classifier that associates the extracted instances to ontology classes. When IE techniques [6] are applied, the generated classifier is based on rules in the form of [22]:

$$r_j = (\text{Condition}) \Rightarrow y_j \quad (2)$$

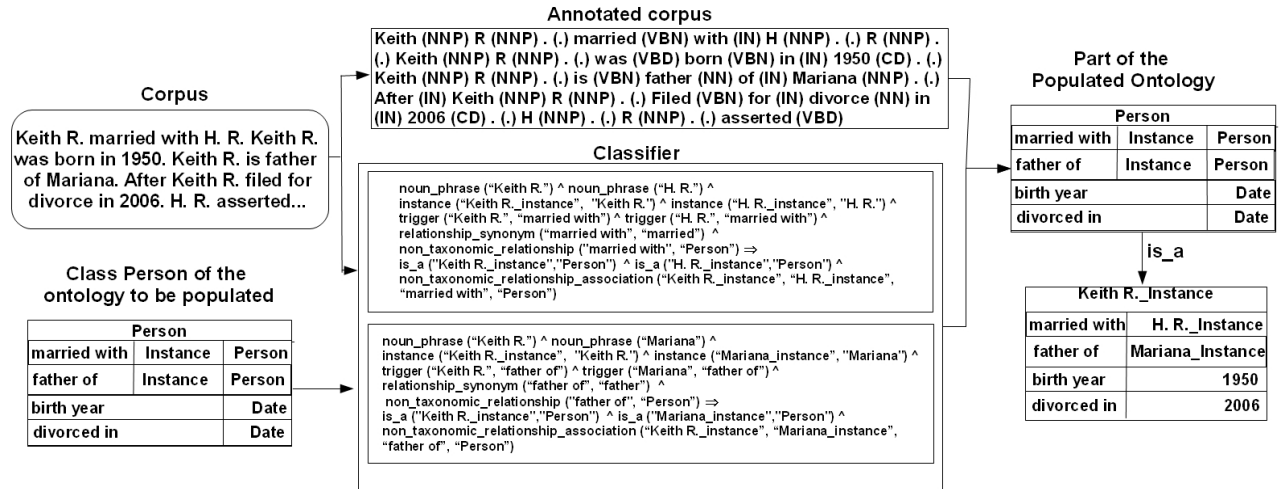


Figure 2. An example illustrating the application of a generic process for automatic ontology population.

The left hand side of the rule is the condition, composed a set of attributes:

$$(A_1 \text{ op } v_1) \wedge (A_2 \text{ op } v_2) \wedge (A_3 \text{ op } v_3) \wedge \dots (A_k \text{ op } v_k) \quad (3)$$

where (A_j, v_j) is a pair attribute-value and op is a relational operator $\{=, \neq, <, >, \leq, \geq\}$. For instance, "noun_phrase ("Keith R.") ^ noun_phrase ("H. R.") ^ instance ("Keith R._instance", "Keith R.") ^ instance ("H. R._instance", "H. R.") ^ trigger ("Keith R.", "married with") ^ trigger ("H. R.", "married with") ^ relationship_synonym ("married with", "married") ^ non_taxonomic_relationship ("married with", "Person")" is a condition that indicates that noun phrases preceded or followed by a trigger are instances. The right hand side of the rule is the class (v_i) to which the instance belongs. For instance, "is_a ("Keith R._instance", "Person") ^ is_a ("H. R._instance", "Person") ^ non_taxonomic_relationship_association ("Keith R._instance", "H. R._instance", "married with", "Person")" is a rule that indicates that class and non taxonomic relationship to which the instance belongs. Fig. 3 shows a classifier based on rules generated from the corresponding corpus and the class "Person" of the ontology to be populated.

When ML techniques [2] are chosen, a manually built training set of positive and negative samples of classified instances is used to generate the classifier. If its classification performance is not suitable a new training set can be constructed to generate a new classifier.

The "Classification of Instances" phase uses the classifier generated in the "Construction of a Classifier" phase to associate instances to their corresponding classes, properties and non-taxonomic relationships. This phase has an annotated corpus, a classifier and an ontology as inputs and outputs a populated ontology. For example, Fig. 3 shows the class "Person" of the ontology in legal domain with the instances: {married_with("Keith R.", "H. R."), father_of("Keith R.",

"Mariana"), birth_year("Keith R.", "1950"), divorced_in ("Keith R.", "2006")}.

IV. AUTOMATIC ONTOLOGY POPULATION APPROACHES

In this section some of the main techniques of the state of the art of AOP are analyzed. The particular solutions adopted to approach each one of the generic phases of the proposed OP process (Fig. 1) are highlighted and their positive aspects and limitations are discussed.

The Fleischman and Hovy [11] approach classifies names of people in eight pre-defined subclasses: athlete, politician, artist, lawyer, clergy, doctor, scientist and police. The "Identification of Candidate Instances" phase applies NLP techniques such as Tokenization, Stemming, Part of Speech Tagging and Named Entity Recognition on a corpus of news stories. The "Construction of a Classifier" phase is accomplished through the application of the decision tree and neural network supervised machine learning techniques. The "Classification of Instances" phase occurs through the application of the induced classifier. In the reported experiments, the classifier with highest precision (70.4%) was the decision tree using algorithm C4.5 [2].

In the Evans [9] approach, the "Identification of Candidate Instances" phase applies NLP techniques such as Tokenization, Stemming, Part of Speech Tagging and Named Entity Recognition on a corpus with documents in the areas of law, psychology, arts and literature. After the identification of named entities queries are submitted using Hearst patterns [16] to a search engine, Google, to identify hyperonyms. An hyperonym is a word that has a wider meaning than its hyponym (a more specific meaning term). The "Construction of a Classifier" phase is accomplished through the application of IE techniques and hierarchical clustering a unsupervised machine learning technique, where the hyperonyms are grouped to obtain the "clusters". Wordnet [10] is used to choose the label of each "cluster". In the "Classification of Instances" phase, instances represented as feature vectors are

compared with corresponding centroids of each cluster to determine the class where the instance will be classified. Experiments report a precision of 92.25%.

The Tanev and Magnini [23] approach aims at populating ontology with entities named people and geographic locations. It proposed an approach called "class-example". The "Identification of Candidate Instances" phase applies NLP techniques to identify candidate instances from the corpus. The "Construction of a Classifier" phase is accomplished through the application of supervised machine learning techniques. The training set consists of lists of instances that can be acquired from ontology or gazetteer lists. The algorithm learns from a training set with a single feature vector called a syntactic model of the class. The "Classification of Instances" phase applies the induced classifier, which compares the vector of syntactic features of the class with the instance one. To evaluate the performance of the approach based on class-example it was compared to the approaches based on patterns and based on class-word. In the reported experiment this three solutions obtained 65%, 18% and 32% for precision respectively.

In Cimiano and Volker [4] approach, the "Identification of Candidate Instances" phase applies NLP techniques to identify candidate instances from the touristic corpus. The "Construction of a Classifier" phase is performed by applying an unsupervised machine learning which is based on Harry's distributional hypothesis [15] and on the vector space model [21]. The context is represented by syntactical features (for example, syntactic functions of the word in a sentence: subject, object, etc.) which are represented by a context vector which is constructed for each class and each instances. The "Classification of Instances" phase applies the induced classifier, which identifies the class to each the instance should be attached by comparing the context vector. Experiments report a precision of 36.82%.

The Etzioni et. al. [8] approach aims at populating ontology. The "Identification of Candidate Instances" phase applies NLP techniques to identify candidate instances from the corpus. The "Construction of a Classifier" phase is performed by applying IE combined with supervised machine learning techniques. Queries to search engines like Google, Alta Vista and Fast are automatically formulated using Hearst patterns [16], for calculating the PMI ("Pointwise Mutual Information") between words and phrases. The PMI is the number of times the instance appears in the sentence divided by the number of times the instance appears alone. For example, the term "St. Louis" obtained 4 for PMI, because it appears 20 times in the sentence "St. Louis is a city" and five times alone. Pairs of instances and their respective classes with the greatest values of PMI are used in training set to generate the bayesian classifier. The "Classification of Instances" phase applies the induced Bayesian classifier, which classifies instances in ontology classes. Etzioni et. al. make several experiments measuring of "recall" and "precision". In the reported experiments, the best result showed 90% and 65% of precision and recall respectively.

In Cimiano et. al. [5] approach, the "Identification of Candidate Instances" phase applies NLP techniques to identify candidate instances from the corpus. The "Construction of a

Classifier" phase is performed by applying IE techniques that uses Hearst patterns [16] and other patterns from the observation of the domain expert upon the corpus. The instance is then classified in the "Classification of Instances" phase in accordance with the principle of maximum evidence. For example, the Nile appears 10 times as a country, appears 50 times the Nile as a river in the corpus, then from the principle of maximum evidence, will be classified as a River. In the reported experiments a precision was of 74.37%.

The Karkaletsis et. al. [17] approach aims at populating ontology. The approach is semi-automatic with the intervention of the domain expert. The "Identification of Candidate Instances" phase applies Named Entity Recognition and Parsing NLP techniques on a Biomedicine corpus. The "Construction of a Classifier" phase is accomplished through the application of IE techniques, where the domain expert identifies the lexical syntactic patterns, combined with a clustering techniques. The "Classification of Instances" phase occurs through the application of the classifier and then the domain expert evaluates the classified instances and generates the populated ontology.

ROSA [13] system automatic instantiate frames (structures of knowledge representation ontologies precursor) for the construction of internal representations of information software describing artifacts. The "Identification of Candidate Instances" phase applies NLP techniques in software descriptions in natural language. The ROSA system consists basically of a retrieval and classification mechanism. The classification engine catalog software components in a knowledge base of frames through the software functionality described in natural language, or instance frames with the internal representation of the functionality of software components. The mechanism makes the similarity analysis between the user's query and the internal representation of the software components stored in frames. The "Construction of a Classifier" phase is accomplished through the application of IE techniques using semantic cases. The semantic cases markers are words in natural language that identify possible instances of slots of the frames. For example, a frame slot with "location", the words that help to identify "location" are "in", "at", "into", "on", "onto", "over" and "within", which are markers of the semantic case "location". The "Classification of Instances" phase occurs through the application of classification mechanism.

The Ruiz-Martinez et. al. [19] approach aims at populating ontology. The "Identification of Candidate Instances" phase applies Named Entity Recognition and Gazetteer Lists NLP techniques on a tourism corpus. The "Construction of a Classifier" phase is performed by applying IE techniques with manual construction of lexical syntactic patterns. The "Classification of Instances" phase occurs through the application of the constructed lexical syntactic patterns, which associate instances to their corresponding ontology classes. In the reported experiments precision was of 92%.

The Carlson et. al. [3] approach aims at populating ontology. The "Identification of Candidate Instances" phase applies NLP techniques to identify candidate instances from the corpus. The "Construction of a Classifier" phase is performed by applying supervised and unsupervised machine learning

techniques. The "Classification of Instances" phase occurs through the application of the induced classifier, which associate instances to their corresponding ontology classes. In the reported experiments precision was of 90%.

ontoX [24] system aims at extracting and classifying instances of properties, but doesn't populate ontology. The "Construction of a Classifier" phase is performed by applying IE techniques with construction of lexical syntactic patterns using keywords. Keywords are words in natural language that identify possible instances of the properties. For example, a property "display size", the keywords that helps to identify "display size" are "inch", "led" and "display". The "Classification of Instances" phase occurs through the application of the constructed lexical syntactic patterns, which associate instances to their corresponding properties. In the reported experiments precision was of 67,8% and a recall was of 68,4%. One of the main limitations is the use of keywords for extraction of instances. Furthermore, it only extracts instances of properties and doesn't instantiate the ontology.

Table 1 shows a comparison of current approaches for OP regarding the techniques employed, supporting tools and reported effectiveness.

TABLE I. ONTOLOGY POPULATION APPROACHES

Approaches / Year	Techniques	Tools	Precision
Carlson et. al. / 2010	NLP, ML	-	94%
Cimiano and Volker / 2005	NLP, ML, IE	Pankow	36,82%
Cimiano et. al. / 2005	NLP, IE	C-Pankow	74,37%
Etzioni et al. / 2005	NLP, ML, IE	KnowItAll	90%
Evans / 2003	NLP, ML, IE	NERO	92,25%
Fleishman and Hovy / 2002	NLP, ML	MenRun	70,4%
Karkaletsis et. al. / 2006	NLP, ML, IE	MPIRO and NLG	-
ontoX / 2007	IE	ontoX	67,8%
Rosa / 1995	NLP, IE	ROSA	-
Ruiz Martinez et al. / 2008	NLP, IE	GATE	90%

According to the performed analysis, all approaches apply the same tasks for AOP: "Identification of Candidate Instances", "Construction of a Classifier" and "Classification of Instances". The "Identification of Candidate Instances" phase applies NLP techniques to identify candidate instances from the corpus. The "Construction of a Classifier" task is performed through the application of ML techniques by the approaches [3] [4] [5] [8] [9] [11] [17] [23], while others [13] [19] [24] apply IE. The "Classification of Instances" task is performed by applying a classifier constructed through induction in the approaches of [3] [4] [5] [8] [9] [11] [17] [23], while others [13] [19] [24] apply syntactic lexical patterns.

The evaluating tools used by the analyzed approaches are proprietary ones, unfortunately not available for testing and comparison purposes.

The effectiveness of the approaches, in terms of the precision percentage shown in Table 1 cannot be considered for comparative purposes among them because each evaluation was conducted using different corpora and ontologies. However the values shown the effectiveness of each approach.

The information source used by the approaches of [3] [5] [8] is the Web. Other approaches [4] [9] [11] [13] [17] [19] [23] [24] use a corpus in a specific domain.

Only the approach of [17] does semi-automatic Ontology Population. Others do it automatically, according to the information reported by authors.

One of the main limitations identified in the approaches is the dependence on a specific domain.

V. EVALUATION

There are typically three ways to evaluate an approach to the AOP. The evaluation can be done measuring the effectiveness of the classification of instances. Other evaluation is based in a comparison with reference ontology. And other evaluation of the populated ontology its use in knowledge-based systems.

An adaption of the classical measures of recall and precision from the field of information retrieval were used for the evaluation of effectiveness [7] considering the number of instances correctly classified.

Precision measures the ratio between the number of instance correctly classified (NICC) and the number of all instance classified (NIC):

$$P = \text{NICC} / \text{NIC} \quad (3)$$

Recall measures the ratio between the number of instance correctly classified (NICC) and the number of all instances in the corpus (NI):

$$R = \text{NICC} / \text{NI} \quad (4)$$

F-measure gives an harmonic mean of precision (P) and recall (R):

$$\text{F-measure} = 2 * P * R / (P+R) \quad (5)$$

The evaluation of instances occurs by comparing the automatically populated ontology with a reference ontology populated by domain experts and knowledge engineers.

About the manual evaluation of instances, one advantage is that the domain experts and knowledge engineers are able to identify whether the ontology is populated automatically good or not, whereas a disadvantage is that the evaluation is subjective and slow.

VI. CONCLUDING REMARKS

This article analyzed the problem and main approaches for Ontology Population, highlighting their positive aspects and limitations, along with a generic process which consists of three phases: "Identification of Candidate Instances",

“Construction of a Classifier” and “Classification of Instances”. The “Identification of Candidate Instances” phase applies NLP or SIR techniques to identify instances of non-taxonomic relationships and properties of an ontology. The “Construction of a Classifier” phase applies IE or ML techniques to build a classifier. The “Classification of Instances” phase uses the classifier generated in the “Construction of a Classifier” phase to associate instances to their corresponding classes, properties and non-taxonomic relationships. One of the main limitations identified in the analyzed approaches [4] [9] [11] [13] [17] [19] [23] is the dependence on a specific domain. Only one by them performs semi-automatic Ontology Population [17].

Future work will develop a domain-independent process. This process proposes a new approach for ontology population using the automatic generation of a classifier from the ontology to be instantiated. One of the main advantages of this proposal is that it is domain-independent because the classifications rules are generated from the ontology during execution time and not a priori. Thus, once the input ontology is defined, the process automatically populates it in a specific domain with instances extracted from documents in natural language. DIPPAOTool a prototype of a tool that support the process is also being developed.

Further work also includes the development of an integrated development environment for ontology learning and population through the composition of DIPPAOTool with other tools constructed by our group in the context of the HERMES project.

ACKNOWLEDGMENT

This work is supported by CNPq, CAPES and FAPEMA, research funding agencies of the Brazilian government.

REFERENCES

- [1] J. Allen, J. Natural Language Understanding. Redwood City, CA: The Benjamin/Cummings Publishing Company, Inc, 1995.
- [2] C. M. Bishop. Pattern Recognition and Machine Learning, Springer, 2006.
- [3] A. Carlson, J. Betteridge, R. Wang, E. Jr. Hruschka and T. Mitchell. Coupled Semi-Supervised Learning for Information Extraction. In: Proceedings of the third ACM international conference on Web search and data mining (WSDM '10). New York, NY, USA, ACM, 2010, pp. 101-110.
- [4] P. Cimiano and J. Volker. Towards large-scale, open-domain and ontology-based named entity classification. In: Proceedings of RANLP'05, Borovets, Bulgaria, 2005, pp. 166-172.
- [5] P. Cimiano, G. Ladwig and S. Staab. Gimme 'the context: Context-driven automatic semantic annotation with C-PANKOW. In Proceedings of the 14th World Wide Web Conference (WWW), 2005, pp. 332-341.
- [6] H. Cunningham. Information Extraction, Encyclopedia of Language and Linguistics, 2nd Edition, 2005.
- [7] K. Dellschaft and S. Staab. On how to perform a gold standard based evaluation of ontology learning. In: Proceedings of the 5th International Semantic Web Conference, Athens. Springer, 2006, pp. 228 – 241.
- [8] O. Etzioni, M. Cafarella, D. Downey, A. M. Popescu, T. Shaked, S. Soderland, D. Weld and A. Yates. Unsupervised named-entity extraction from the web: An experimental study, Artificial Intelligence, 2005, 165(1):91-134.
- [9] R. Evans. A framework for named entity recognition in the open domain. In Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP), 2003, pp. 137-144.
- [10] C. Fellbaum. Wordnet: An Electronic Lexical Database. Cambridge: MIT Press, 1998.
- [11] M. Fleischman and E. Hovy. Fine Grained Classification of Named Entities. In: Proceedings of COLING, Taipei, Taiwan, August, 2002.
- [12] R. Girardi. Guiding Ontology Learning and Population by Knowledge System Goals. In: Proceedings of International Conference on Knowledge Engineering and Ontology Development, Ed. INSTIIC, Valence, 2010, pp. 480 – 484.
- [13] R. Girardi. Using English to Retrieve Software, The Journal of Systems and Software, v. 30, n. 3, 1995, pp. 249-270.
- [14] N. Guarino, C. Masolo and C. Vetere. Ontoseek: Content-based Access to the web, IEEE Intelligent Systems, v. 14(3), 1999, pp. 70-80.
- [15] Z. Harris. Distributional structure. In J.J. Katz, editor, The Philosophy of Linguistics, New York: Oxford University Press, 1995, pp. 26-47.
- [16] M. Hearst. Automated Discovery of Word-Net Relations. In WordNet: An Electronic Lexical Database. MIT Press, 1998.
- [17] V. Karkaletsis, A. Valarakos and C. D. Spyropoulos. Populating ontologies in biomedicine and presenting their content using multilingual generation. Acquiring and Representing Multilingual, Specialized Lexicons: the Case of Biomedicine, Italy, Genoa, 2006.
- [18] R. Porzel and R. Malaka. A Task-based Approach for Ontology Evaluation. In ECAI Workshop on Ontology Learning and Population, Valencia, Spain, 2004.
- [19] J. M. Ruiz-Martinez, J. A. Minarro-Gimenez, L. Guillen-Carceles, D. Castellanos-Nieves, R. Valencia-Garcia, F. Garcia-Sanchez, J. T. Fernandez-Breis and R. Martinez-Bejar. Populating Ontologies in the eTourism Domain. In Proceedings of the 2008 IEEE/WIC/ACM international Conference on Web intelligence and intelligent Agent Technology - Volume 03. Web Intelligence & Intelligent Agent. IEEE Computer Society, Washington, DC, December 09 - 12 - 2008, pp. 316-319.
- [20] S. Russel and P. Norvig. Artificial Intelligence: A Modern Approach, Prentice-Hall, 1995.
- [21] G. Salton and C. Buckley. Term Weighting Approaches in Automatic Text Retrieval. Cornell University, 1987.
- [22] P. N. Tan, M. Steinbach and V. Kumar. Introduction to Data Mining, Pearson Addison Wesley, 2005.
- [23] H. Tanev and B. Magnini. Weakly Supervised Approaches for Ontology Population. In: Proceedings of of the 11th Conference of the European Chapter of the Association for Computational Linguistics (EACL), 2006, pp. 17-24.
- [24] B. Yildiz and S. Miksch. ontoX - a method for ontology-driven information extraction. In Proceedings of the international conference on Computational science and its applications (ICCSA'07) - Volume Part III, Osvaldo Gervasi and Marina L. Gavrilova (Eds.), Vol. III. Springer-Verlag, Berlin, Heidelberg, 2007, pp. 660-673.