Proceedings of Symposium in Information and Human Language Technology. Uberlândia, MG, Brazil, October 2–5, 2017.

A Rule-based Semantic Annotator: Adding top-level ontology Tags

Guidson Coelho de Andrade[1], **Alcione de Paiva Oliveira**[1] **and Alexandra Moreira**[1]

[1]Departamento de Informática – Universidade Federal de Viçosa (UFV)
Campus Universitário – 36.570-900 – Viçosa – MG – Brazil

{guidson.c.andrade, alcione}@gmail.com, xandramoreira@yahoo.com.br

Abstract. *Understanding natural language texts is a simple task for human beings, but despite recent advances, it is still a challenge for computational devices. An important step in allowing machines to understand texts in natural language is to annotate lexemes with semantic information. Semantic information has several levels and aspects, but a type of semantic annotation that has the ability to help determine the context of the statement is the ontological information. However, annotating texts according to an ontology is still a task that requires time and effort from annotators trained for this purpose. The goal of the project is to assist in the semantic enrichment of texts, through a rule-based annotator. Given an entry in the format required by the annotator, the tool returns a document annotated according to the concepts proposed by the SUMO ontology. The project consists in elaborating a semantic annotator based on rules that is able to annotate a corpus using the selected top-level ontologies.*

1. Introduction

Assigning semantic information to lexemes is a task that has made significant progress recently, mainly due to the increase in computational power and to the availability of large linguistic *corpora* to train automatic learning tools. Nowadays, there are commercial devices such as smartphones or Amazon® Echo that are capable of answering questions made in natural language. In order for these devices to function properly, some semantic information must be attributed to the utterances, even if implicitly through statistical analysis. Another way to aid in the understanding of texts by computer devices is to explicitly add semantics to textual information by annotating lexemes with semantic information.

There are different annotation granularities that range from associating a label to a full text to associating a label with each phrase or even word [Leech 1997]. Semantic annotation is an annotation that attempts to unveil the meaning of the things being marked [Reeve and Han 2005]. Semantic annotation searches for text elements and classifies them according to their meaning in the fragment in which they are inserted [Mitkov 2005]. Semantics has several levels and aspects, but a type of semantic information that has the ability to help determine the context of a statement is the ontological information. The term ontology can be defined as a specification of a conceptualization [Gruber et al. 1993]. In other words, it is a description of concepts of existing entities in the world and relationships that exist between these entities [Uschold and Gruninger 1996]. Ontology studies the various entities that exist, a description of types and structures of things, their properties, events, relationships and processes throughout the real world [Guarino 1998]. The ontologies are used to classify

the objects of some domain, according to some pre-established criteria [Maedche 2012]. The annotation based on ontologies, although not much explored, could provide, under a certain level of abstraction, contextual information to the annotated textual object [Handschuh and Staab 2003]. However, annotating texts according to an ontology is still a task that requires time and effort from annotators trained for this purpose, and it is still commonly elaborated through manual work [Pustejovsky and Stubbs 2012]. The exhaustive work caused by this task is responsible for the lack of ontology annotated corpora. A tool capable of semantically annotating texts based on ontology would be very useful and would help increase the availability of corpora annotated with this type of information.

The objective of the research presented in this article was to construct a semantic annotator based on rules capable of annotating the terms of a given corpus under the concepts of a top-level ontology. It uses the concepts of the chosen ontology level and classifies the terms of the corpus to annotate them according to the ontology. The tool developed is domain independent but has been implemented with focus on the English language.

This paper is organized as follows: the next section presents the work previously developed that are related to this research; Section 3 describes the materials and methods applied in the research; Section 4 presents the results obtained; and Section 5 presents the final remarks.

2. Related work

The semantic annotation field is quite active, but most of the work deals with annotation of semantic roles. Here we will discuss some recent work dealing with ontological annotation.

[Asooja et al. 2016] developed a system to automatically annotate texts of the regulatory sector for different industries using the semantic frames via FrameNet, which is, in a certain sense, a lexical ontology. The application of the FrameNet lexical base contributed to the increased performance of its results. The system also made use of POS annotation and n-grams. The difference in relation to our work is that we choose to use a formal ontology rather than a lexical ontology (the distinction is that lexical ontology has inspiration in what is enunciated rather than in what exists). Another distinct point of the work of Asooja et al. is that they used classification by means of statistical techniques while we used rules.

[Alec et al. 2016] proposed an ontology-driven approach for semantic annotation of documents from a corpus where each document describes an entity of a same domain. The focus of the work was more to annotate documents rather than the words of the documents. In addition, the researches used domain ontologies instead of a top-level ontology.

[Moreira et al. 2016] proposed a system that extracts the terms of a text and links them to an ontology (SUMO ontology in that case). The system could be used to annotate the text but was not used for this purpose. In addition, the system analyzes only terms that originate from noun phrases, which is a more limited scope than the current research.

[Pham et al. 2016] presented a domain-independent approach to automatic semantic labeling that uses machine learning techniques. Similarly to our proposal the domain-

independent feature was the novelty of their approach. Unlike our approach which is a rule-based method, the authors used similarity metrics as features to compare against labeled domain data and learns a matching function to infer the correct semantic labels for data. They also focused on domain ontology rather than on top-level ontology.

3. Materials and Methods

The Suggested Upper Merged Ontology (SUMO) [Pease et al. 2002] was the top-level ontology chosen for this project. Its choice was based on being an ontology with a certain degree of maturity, with a broad scope and for being well formalized. SUMO was first released in December 2000 and defines a hierarchy of classes, rules, and relationships [Niles and Pease 2001]. It is intended to be an ontology that underpins a variety of computer information processing systems [Pease et al. 2002]. Although it is an ontology that addresses some domains, in our work we focus only on top-level concepts, because we believe that this first step is essential for a later annotation focused on a specific domain.

This work was developed using the concepts of the first three levels of SUMO ontology. The top level of the SUMO ontology contains 12 classes distributed in the three levels, as shown in Figure 1. The first level displays the root class named *Entity*. The *Physical* and *Abstract* classes compose the second level. And finally, on the third level of the ontology there are the classes, *Object*, *Process*, *Quantity*, *Attribute*, *Set Or Class*, *Relation*, *Proposition*, *Graph* and *Graph Element*. Each class has a formal definition, allowing to distinguish which entity can belong to the class. The semantic annotator was constructed by creating rules to assign the lexemes of a text to their respective ontological class in the SUMO ontology, some examples of rules are provided in Figure 2.

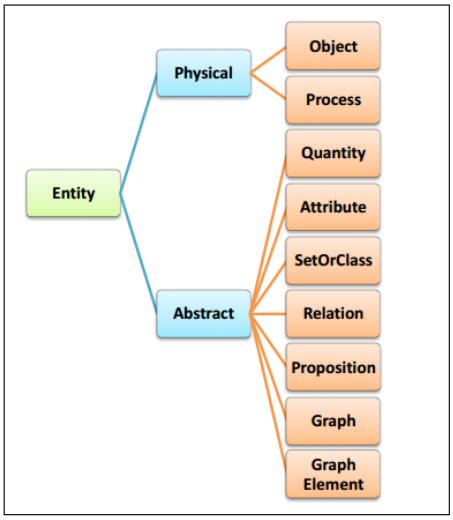

Figure 1. Top layers of the SUMO Ontology

The Open American National Corpus (OANC) [Ide and Suderman 2004] was selected to carry out training and tests of the annotator. The linguistic diversity of the

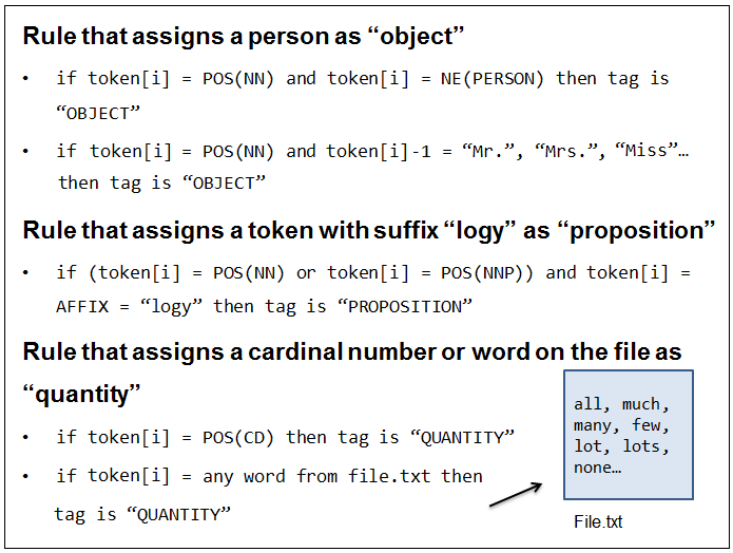

Figure 2. Rules examples

Open American National Corpus allows expressing a wide range of language expressions and covering the largest number of words in American English. OANC is a corpus composed of 5 million words derived from various textual and oral genres of American English [Ide and Suderman 2004]. It is free of charge and available for download. It is annotated according to structural markup, sentence boundaries, part of speech, noun chunks and verb chunks, which justifies the choice of the corpus for the application [Ide and Suderman 2004]. The annotation provided by the corpus served as the basis for the construction of the rules of this work.

Due to the massive amount of documents, it was necessary to make a snippet of the corpus to turn the application development more manageable. The sub-corpus chosen was initially the text entitled "Who Killed Martin Lutter King?". The sample was used to illustrate the procedure adopted by the application to perform a properly annotation. The document in .xml format was extracted containing the annotations provided by the corpus. The .xml document, as well as all the corpus files, are in the Linguistic Annotation Format (LAF) (ISO 24612) standard for creating annotated corpus. In order to process the document it was necessary to normalize it, excluding paragraph markings, white space, headers and structural tags. The output of the normalization was a .txt document containing only annotated sentences.

The file generated in the previous phase went through further transformations. An important information for the application being developed is the named entity annotation, however the OANC does not provide this type of annotation. In order to add this layer of annotation to the corpus it was used the Stanford NER, a named entities annotator. Stanford NER is a annotator created by the Stanford Natural Language Processing Group, and it annotates entries under the categories "PERSON", "ORGANIZATION" and "LOCATION", using the Conditional Randon Field (CRF) approach[Finkel et al. 2005]. The

outcome of this phase was a .txt file having the annotations and the format required for the development of the semantic annotator.

The semantic annotator proposed has three phases, formatting for annotation, annotation and post annotation. The formatting phase formats the input document into a structure capable of being interpreted by the annotator. The annotation is the step that marks the elements present in the text according to a SUMO ontology category. Finally, post annotation uses the already annotated structure to create the annotated .txt document. The details of each phase will be described in the following paragraphs.

A document consists of a series of sentences, which in turn is composed of a series of tokens. Each sentence token has become a dictionary entry where the key is the number of the sentence and the value of the entry is a list of pair $< token, attributes >$. The $attributes$ is a set of syntactic and semantic information about the token. Figure 3 shows a sentence with the annotations and Figure 4 shows the dictionary structure.

<s><tok base="last" msd="JJ" ne="O">Last</tok> <tok base="week" msd="NN" ne="O">week</tok><tok base="," msd="," ne="O">,</tok> <tok base="a" msd="DT" ne="O">a</tok> <tok affix="s" base="memphi" msd="NNP" ne="LOCATION">Memphis</tok> <tok base="jury" msd="NN" ne="O">jury</tok> <tok affix="ed" base="find" msd="VBD" ne="O">found</tok> <tok base="that" msd="DT" ne="O">that</tok> <tok base="restaurant" msd="NN" ne="O">restaurant</tok> <tok base="owner" msd="NN" ne="O">owner</tok> <tok base="loyd" msd="NNP" ne="PERSON">Loyd</tok> <tok affix="s" base="jower" msd="NNP" ne="PERSON">Jowers</tok> <tok affix="ed" base="be" msd="VBD" ne="O">was</tok> <tok affix="ed" base="involve" msd="VBN" ne="O">involved</tok> <tok base="in" msd="IN" ne="O">in</tok> <tok base="a" msd="DT" ne="O">a</tok> <tok base="conspiracy" msd="NN" ne="O">conspiracy</tok> <tok base="to" msd="TO" ne="O">to</tok> <tok base="kill" msd="VB" ne="O">kill</tok> <tok base="martin" msd="NNP" ne="PERSON">Martin</tok> <tok base="luther" msd="NNP" ne="PERSON">Luther</tok> <tok base="king" msd="NNP" ne="PERSON">King</tok> <tok base="jr" msd="NNP" ne="PERSON">Jr</tok><tok base="." msd="." ne="O">.</tok></s>

Figure 3. Sentence previous annotation sample.

After the formatting for annotation phase, the actual annotation phase began. The annotation phase consists of applying rules that evaluate whether a token belongs to an ontological category. The rules evaluate several aspects of the token, such as its affixes, POS, named entity annotation, neighborhood tokens and occurrence in gazetteers lists. When the token is classified under a given rule, the class tag is added to the attribute list of the token. The classification occurs inversely, from the third to the first level, because if the token is classified under a class of the third level, it is already possible to say its second and first level.

At the end of the classification the token receives three labels (so1stl, for the first level of the SUMO ontology, so2ndl for the second level of the SUMO ontology, and so3rdl for the third level of the SUMO ontology). The classification adds the tags to the attributes list of the token, receiving the "CLASS NAME" if the rule applies to the token or "O" if the rules does not apply to it. If applicable to the first level of the SUMO ontology the annotator adds the "ENTITY" tag. At the second level the annotator may mark the token as "PHYSICAL" or "ABSTRACT". At the third-level annotator tags are "OBJECT", "PROCESS", "QUANTITY", "ATTRIBUTE", "SET OR CLASS", "RELATION", "PROPOSITION", "GRAPH" and "GRAPH ELEMENT".

The tokens of the document receive annotation related to the ontology, changing in the form exemplified by Figure 5. The third step restructures the entire list of sentences in a document annotated according to the ISO format Linguistic Annotation Format (LAF)

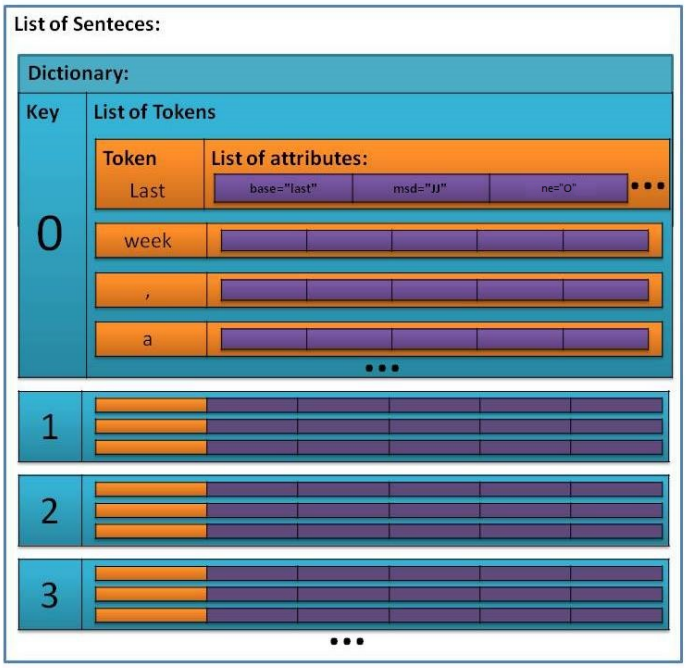

Figure 4. Dictionary structure.

(ISO 24612) model. This phase is necessary because it is important that the document outputted by the annotator be in a standard format that can be used by other applications.

<s><tok base="last" msd="JJ" ne="O" so1stl="ENTITY" so2ndl="ABSTRACT" so3rdl="ATRIBUTE">Last</tok> <tok base="week" msd="NN" ne="O" so1stl="ENTITY" so2ndl="O" so3rdl="O">week</tok> <tok base="," msd="," ne="O" so1stl="O" so2ndl="O" so3rdl="O">,</tok> <tok base="a" msd="DT" ne="O" so1stl="O" so2ndl="O" so3rdl="O">a</tok> <tok affix="s" base="memphi" msd="NNP" ne="LOCATION" so1stl="ENTITY" so2ndl="PHYSICAL" so3rdl="OBJECT">Memphis</tok> <tok base="jury" msd="NN" ne="O" so1stl="ENTITY" so2ndl="O" so3rdl="O">jury</tok> <tok affix="ed" base="find" msd="VBD" ne="O" so1stl="O" so2ndl="O" so3rdl="O">found</tok> <tok base="that" msd="DT" ne="O" so1stl="O" so2ndl="O" so3rdl="O">that</tok> <tok base="restaurant" msd="NN" ne="O" so1stl="ENTITY" so2ndl="O" so3rdl="O">restaurant</tok> <tok base="owner" msd="NN" ne="O" so1stl="ENTITY" so2ndl="O" so3rdl="O">owner</tok> <tok base="loyd" msd="NNP" ne="PERSON" so1stl="ENTITY" so2ndl="PHYSICAL" so3rdl="OBJECT">Loyd</tok> <tok affix="s" base="jower" msd="NNP" ne="PERSON" so1stl="ENTITY" so2ndl="PHYSICAL" so3rdl="OBJECT">Jowers</tok> <tok affix="ed" base="be" msd="VBD" ne="O" so1stl="O" so2ndl="O" so3rdl="O">was</tok> <tok affix="ed" base="involve" msd="VBN" ne="O" so1stl="O" so2ndl="O" so3rdl="O">involved</tok> <tok base="in" msd="IN" ne="O" so1stl="O" so2ndl="O" so3rdl="O">in</tok> <tok base="a" msd="DT" ne="O" so1stl="O" so2ndl="O" so3rdl="O">a</tok> <tok base="conspiracy" msd="NN" ne="O" so1stl="ENTITY" so2ndl="O" so3rdl="O">conspiracy</tok> <tok base="to" msd="TO" ne="O" so1stl="O" so2ndl="O" so3rdl="O">to</tok> <tok base="kill" msd="VB" ne="O" so1stl="O" so2ndl="O" so3rdl="O">kill</tok> <tok base="martin" msd="NNP" ne="PERSON" so1stl="ENTITY" so2ndl="PHYSICAL" so3rdl="OBJECT">Martin</tok> <tok base="luther" msd="NNP" ne="PERSON" so1stl="ENTITY" so2ndl="PHYSICAL" so3rdl="OBJECT">Luther</tok> <tok base="king" msd="NNP" ne="PERSON" so1stl="ENTITY" so2ndl="PHYSICAL" so3rdl="OBJECT">King</tok> <tok base="jr" msd="NNP" ne="PERSON" so1stl="ENTITY" so2ndl="PHYSICAL" so3rdl="OBJECT">Jr</tok> <tok base="." msd="." ne="O" so1stl="O" so2ndl="O" so3rdl="O">.</tok> </s>

Figure 5. Sentence annotated with the ontological categories.

4. Results

In this section we present the results of the test conducted with the annotator on a text sample to illustrate the its performance. The system has been tested with 39 sentences and 908 tokens from the text sample mentioned in the previous chapter after it being manually

annotated. In addition is shown the confusion matrix, precision and recall measurements performed for each ontological class, assuming the annotation results from the chosen sub-corpus.

Because of the huge size of the entire chosen corpus used to build the rules, it was necessary to select a sample text to perform the test. The test was conducted by manually adding tags to the documents according to the top-level ontology and then comparing it with the same sample annotated by the application. Although it is a limited fragment of the corpus, the text exemplifies how the annotator would perform if the corpus was already manually annotated.

Table 1. Confusion matrix. On the vertical are the classes that should be assigned to the tokens and horizontally those that were assigned by the annotator.

	OBJEC	PROCE	QUANT	ATTRI	SETCL	RELAT	PROPO	GRAPH	GRAEL	NONON
OBJEC	97	0	0	0	0	0	0	0	0	0
PROCE	0	14	0	0	0	0	0	0	0	0
QUANT	0	0	15	0	0	0	0	0	0	0
ATTRI	0	0	0	46	0	0	0	0	0	0
SETCL	4	0	0	0	1	0	0	0	0	0
RELAT	0	0	0	0	0	0	0	0	0	0
PROPO	0	0	0	0	0	0	0	0	0	0
GRAPH	0	0	0	0	0	0	0	0	0	0
GRAEL	0	0	0	0	0	0	0	0	0	0
NONON	69	14	7	0	0	0	0	0	0	641

Table 1 shows the confusion matrix for the third-level classes. A special class, called “NO ONTOLOGY” was added to handle tokens that did not fit into any class. The vast majority of tokens were annotated correctly. But the result was not good with the terms related to the class “SET OR CLASS”. A minority of tokens related with the special class “NO ONTOLOGY” were misclassified. Note that all elements of this class that were erroneously classified ended up being classified into class ”OBJECT”. This shows that the rule is not distinguishing properly when the concept denoted by a token focuses more on the aspect of the elements than on the parts.

The annotation accuracy relative to the third level of the SUMO ontology was 89.65%. As one can infer from the confusion matrix the precision, recall and F1 measures had good results except for the class “SET OR CLASS”.

```
-----PRECISION-----

OBJECT PRECISION: 0.5705882352941176
PROCESS PRECISION: 0.5
QUANTITY PRECISION: 0.6818181818181818
ATTRIBUTE PRECISION: 1.0
SET_OR_CLASS PRECISION: 1.0
RELATION PRECISION: 0.0
PROPOSITION PRECISION: 0.0
GRAPH PRECISION: 0.0
```

```
GRAPH_ELEMENT PRECISION: 0.0
NO ONTOLOGY PRECISION: 1.0


-----RECALL-----

OBJECT RECALL: 1.0
PROCESS RECALL: 1.0
QUANTITY RECALL: 1.0
ATTRIBUTE RECALL: 1.0
SET_OR_CLASS RECALL: 0.2
RELATION RECALL: 0.0
PROPOSITION RECALL: 0.0
GRAPH RECALL: 0.0
GRAPH_ELEMENT RECALL: 0.0
NO ONTOLOGY RECALL: 0.8768809849521204


-----F1 MEASURE-----

OBJECT MEASURE: 0.7265917602996255
PROCESS MEASURE: 0.6666666666666666
QUANTITY MEASURE: 0.8108108108108109
ATTRIBUTE MEASURE: 1.0
SET_OR_CLASS MEASURE: 0.33333333333333337
RELATION MEASURE: 1.0
PROPOSITION MEASURE: 1.0
GRAPH RECALL: 1.0
GRAPH_ELEMENT MEASURE: 1.0
NO ONTOLOGY MEASURE: 0.934402332361516
```

The statistical results provided in this section refers only to the text sample and it does not apply to the corpus. The sub-corpus was used only to exemplify the behavior of the annotator comparing to a manually annotated text. To verify the overall metrics of the corpus it would be necessary to hand-annotate all files and afterwards compare them with the rule-annotated documents generated by the application.

5. Conclusions

Semantic annotation allows data to be interpreted by applications in such way that machines can capture the underlying meaning of an utterance. However, annotating documents to help express aspects of their semantic meaning is still challenging, due to the lack of applications that assist the task. Notably, there is some difficulty of finding tools capable of executing semantic annotation in text documents using ontological concepts, this was the main reason for the development of this research. Manual annotation is a task that takes time and knowledgeable staff to carry it out, and the proposal of a rules-based annotator can be of great help.

Therefore, the proposal of this research was the creation of a tool that would aid in the process of semantic annotation based on top-level ontological classes. The tool makes use of a set of rules elaborated according to the concepts described by the ontology and by making use of previous annotations layers provided by the corpus.

Although the experiment described in this paper only used a single document to demonstrate viability of the proposal, it is possible to apply the same technique to the whole OANC. The annotation of the whole corpus helps to enrich it in an ontological dimension, so the text files can be used in futures researches on the semantic annotation field.

The importance of this work is the possibility of increasing the number of annotated corpus with ontological information, which may facilitate the training annotators based on supervised machine learning techniques, enabling a new generation of semantic annotators with higher performance and accuracy.

Acknowledgments.

This research is supported in part by the funding agencies FAPEMIG, CNPq, and CAPES.

References

Alec, C., Reynaud-Delaître, C., and Safar, B. (2016). An ontology-driven approach for semantic annotation of documents with specific concepts. In *International Semantic Web Conference*, pages 609–624. Springer.

Asooja, K., Bordea, G., and Buitelaar, P. (2016). Using semantic frames for automatic annotation of regulatory texts. In *International Conference on Applications of Natural Language to Information Systems*, pages 384–391. Springer.

Finkel, J. R., Grenager, T., and Manning, C. (2005). Incorporating non-local information into information extraction systems by gibbs sampling. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, pages 363–370. Association for Computational Linguistics.

Gruber, T. R. et al. (1993). A translation approach to portable ontology specifications. *Knowledge acquisition*, 5(2):199–220.

Guarino, N. (1998). Formal ontology and information systems. In *Proceedings of FOIS*, volume 98, pages 81–97.

Handschuh, S. and Staab, S. (2003). *Annotation for the semantic web*, volume 96. IOS Press.

Ide, N. and Suderman, K. (2004). The american national corpus first release. In *LREC*. Citeseer.

Leech, G. (1997). *Introducing corpus annotation*. Addison Wesley Longman.

Maedche, A. (2012). *Ontology learning for the semantic web*, volume 665. Springer Science & Business Media.

Mitkov, R. (2005). *The Oxford handbook of computational linguistics*. Oxford University Press.

Moreira, A., Lisboa-Filho, J., and Oliveira, A. P. (2016). Automatic ontology generation for the power industry the term extraction step. In *Proceedings of the 21 International Conference on Applications of Natural Language to Information Systems*, pages 415–420. Springer.

Niles, I. and Pease, A. (2001). Towards a standard upper ontology. In *Proceedings of the international conference on Formal Ontology in Information Systems-Volume 2001*, pages 2–9. ACM.

Pease, A., Niles, I., and Li, J. (2002). The suggested upper merged ontology: A large ontology for the semantic web and its applications. In *Working notes of the AAAI-2002 workshop on ontologies and the semantic web*, volume 28.

Pham, M., Alse, S., Knoblock, C. A., and Szekely, P. (2016). Semantic labeling: a domain-independent approach. In *International Semantic Web Conference*, pages 446–462. Springer.

Pustejovsky, J. and Stubbs, A. (2012). *Natural language annotation for machine learning*. O'Reilly Media, Inc.

Reeve, L. and Han, H. (2005). Survey of semantic annotation platforms. In *Proceedings of the 2005 ACM symposium on Applied computing*, pages 1634–1638. ACM.

Uschold, M. and Gruninger, M. (1996). Ontologies: Principles, methods and applications. *The knowledge engineering review*, 11(02):93–136.