Extracting semantic information from patent claims using phrasal structure annotations

Danilo S. Carvalho
and Felipe M. G. França
Systems Engineering and Computer Science Program - COPPE
Universidade Federal do Rio de Janeiro
Rio de Janeiro, Brazil

Priscila M. V. Lima
Instituto Tércio Pacitti (NCE)
Universidade Federal do Rio de Janeiro
Rio de Janeiro, Brazil

***Abstract*—The rapid change of trading values from tangible assets to Intelectual Property has put both businesses and academia in a race to acquire and protect the rights to exploit such property. This is mainly accomplished in the form of patent issuing by the governments, being time consuming and complicated due to the vast amount of documents that need to be analyzed in order to assert the novelty or validity of a patent application. Patent information retrieval research is thus growing quickly to support document analysis across multiple domains and information systems. One of the big challenges in patent analysis is the identification of the elements of innovation (concepts, processes, materials) and the relations between them, in the patent text. This paper presents a method for extracting semantic information from patent claims by using semantic annotations on phrasal structures, abstracting domain ontology information and outputting ontology-friendly structures to achieve generalization. An extraction system built upon the method is briefly evaluated on a document sample from INPI, the Brazilian patent office, a challenging information source.**

I. Introduction

In the recent years, companies and governments have taken part of a fast transition of trading values: from tangible assets to the *Intellectual Property* concept, with regulations trying to follow the pace of change. Developing innovative designs and processes is becoming an increasingly important task for businesses and academia. However, the management of innovation related information constitutes a very difficult task, which involves the analysis of a vast amount of legal and academic documents.

Patent information retrieval is a way of facilitating such tasks by obtaining the most relevant parts of patent documents, e.g., author and subject, and by organizing them in queryable *knowledge bases* for easy access. Unfortunately, these documents are mostly written in natural language, which poses a big challenge to correct identification of relevant patent parts, specially novelty terms. Some organizations have taken initiatives on making patent data available on the web, like EPS [1], epoline [2](Europe) and Google's USPTO public downloads [3] (United States). Additionally, formats are still not consistent and many other patent offices around the world do not publish their documents online or do so in an unstructured manner, e.g., scanned paperforms in PDF format. A coherent method of retrieval is imperative to integrate data from different offices.

To optimize the retrieval, an possible strategy is to focus the analysis only on the patent claims. A patent claim describes the concept, process or material that is the subject of legal protection in a structured and more precise language than the rest of the patent document. A claim can be *independent*, when it declares a subject of protection, or *dependent*, when it details a previously declared subject. The analysis of a claim comprises the identification and linking of the subject to its details, enabling content-based patent querying and the comparison of patent scopes.

Ghoula et al. [1] described a method for generating semantic annotations on patent texts, using the document structure and a multilevel ontology annotation scheme, supported by a combination of NLP techniques. Although this approach is fast and well aligned with a web semantic perspective, it depends on structured documents and on an existing domain ontology for extracting information from the patent claims. Taduri et al. [2] proposed a patent system ontology, aiming to standardize the representation from different information sources, but initially focusing on US patents office and court records. Yang and Soo [3] presented a method for extracting conceptual graphs from claims using syntactic information and a background ontology, also focusing on the US patents claim structure. For the Portuguese language, Ferreira et al. [4] devised a method for the extraction of non-taxonomic relations between concepts, combining concept extraction using syntactic information with statistical verb-centric approach for relation extraction. Bruckschen et al. [5] presented a rule based method for relation extraction between named entities and Caputo [6] a clustering approach for finding semantic relations in Brazilian patents, using document metadata and summary fields. Important related works also include the method for part-whole domain independent relation extraction, by Girju et al. [7], and the weakly-supervised algorithm for generic pattern relation extraction by Pantel and Pennacchiotti [8], both using semantic annotation learning. This work presents a supervised method for extracting semantic information from patent claims using semantically annotated syntactic structures, which are used to train a weightless neural classifier and syntactic-semantic filters able to annotate unseen claims, i.e. claims not used for training. The extracted information is in the form of RDF triples [9], which can be aligned to any domain ontology. Testing was performed on a set of patent documents

[1]European Publication Server: http://patentinfo.european-patent-office.org/off_pubs/pub_serv/.
[2]http://www.epoline.org.
[3]http://www.google.com/googlebooks/uspto.html

978-1-4799-5618-0/14 $31.00 © 2014 IEEE
DOI 10.1109/BRACIS.2014.17

from INPI[4], the Brazilian patent office, as they present many obstacles from both format and language perspectives.

The remainder of this paper is organized in the following manner: Section II describes the semantic function model used in this work and its utility in the construction of a knowledge base. Section III explains the method for semantic information extraction and the experimental results. Section IV concludes and summarizes the findings.

II. Semantic segmentation

A. Semantic segments

There are many ways to classify the semantic function of a word or phrase in a sentence. *Semantic roles* and *semantic relations* are two widely used classification models. The first identifies each class as an argument of a predicate, while the latter describes relations between concepts expressed in the sentence. Moldovan et al. [10] identifies several relation classes covering a large majority of text semantics. However, semantic functions in different types of text can be better modeled with specific sets of classes. This is the case of patent claims, which can cover a vast number of concept domains, but have a set of common functions, such as patent subject, claim reference and subject characterization.

To attain compatibility with any given set of function classes or ontology, this work uses the concept of **semantic segment**: any subsequence of words in a sentence, to which a concept or relation can be attributed. The semantic segment is a generalization of the *semantic role* and *semantic relation* concepts, where a class represents a semantic function relative to any element inside or outside of the sentence. For example, the class ILUST_REF indicates the number used to reference a specific illustration in the patent document. Figure 1 illustrates a semantically segmented sentence for the Brazilian patent claim (in Portuguese) *"Dispositivo antifurto caracterizado por compreender: 1 Uma caixa blindada e, 2 Um sistema antiarrombamento."* (translation: "*Anti-theft device characterized for comprising: 1 A shielded box and, 2 An anti-burglary system.*").

B. Building a knowledge base

A semantic segment class may represent a relation by itself, but relations between segments are also possible. The collection of all relations found in a text produces a graph $G = (S, R)$, where S is the set of segments and R is the set of relations between segments. When aligned to a domain ontology, this graph can be transformed into a corresponding *knowledge base*, which can, in turn, be stored in a convenient format, e.g., RDF, and queried using popular database systems for knowledge bases, e.g., Virtuoso [11], Apache Jena [12].

III. Semantic Segment Annotation

A. Overview

A patent claim is typically written as a combination of the following semantic functions:

- *Patent subject*: the main subject of legal protection;
- *Reference*: a reference to another claim or patent;
- *Claim reference*: an explicit reference to another claim in the same patent document;
- *Claim reference number*: the number of a referenced claim; it serves as claim ordering and as a unique claim identifier in a document;
- *Subject characterization*: a phrase detailing the patent subject;
- *Patent object*: any other concept, process or material cited in the claim, that is not the subject;
- *Object characterization*: a phrase detailing a patent object;
- *Illustration reference*: a reference to an illustration in the patent document; usually a number or a letter that is a unique identifier for a figure, technical drawing or diagram.

The annotation system must correctly isolate each function from the claims as semantic segments and label them with the corresponding segment classes. The steps to accomplish this are divided in two phases: training and extraction/annotation, which are described next.

B. Training phase

The training phase deals with the structuring of semantic segment data for a machine learning model. In this phase, a set of annotated claims is presented to the system. The annotations are formatted as tree structures and written using the Penn Treebank format [13], which simplifies annotation creation and reading. The structures created in this way are called *semantic segment trees*. Each annotated claim is processed through the following steps:

a) Parsing: In the first step, the claim is analyzed by a *PCFG*[5] *constituent parser*, which outputs the corresponding syntactic tree for the claim sentence. A syntactic tree, also known as phrase structure, is a tree structure where each non-terminal node represents a syntactic constituent (phrasal structure), such as a Noun phrase, Verb phrase or Adjective phrase, and each terminal node is a word from the sentence, represented as a Part-of-Speech. Figure 2 shows the syntactic tree for the claim shown in Figure 1.

Chunking, i.e. Shallow Parsing, was initially chosen as a less costly alternative for parsing. However, the chunks were often too large and thus inadequate for the type of processing done on the next steps, which require finer-grained phrase separation and hierarchical structure for the construction of semantic pattern attributes.

b) Syntactic-semantic aligning: In the second step, the syntactic trees and semantic segment trees are *aligned*. The trees are considered aligned when each semantic segment is paired with a syntactic node that contains all the words from that segment and the minimum of excess words. A segment is *perfectly aligned* if there is a syntactic node that contains only the words in the segment. The alignment is performed by walking depth-first on both trees and comparing the leaves

[4]National Institute for Industrial Property (from Portuguese)

[5]Probabilistic Context-free Grammar

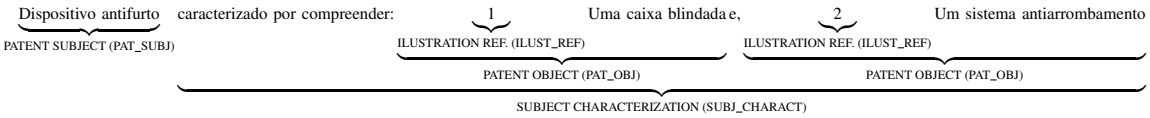

Fig. 1. Semantically segmented sentence. Each segment is a sequence of words that have a semantic function in the sentence. Such function depends on the way the text is being analyzed. In this figure, a patent claim is divided into a *patent subject*, its *characterization*, the *objects* claimed in the patent, and the *references* to their respective *illustrations*.

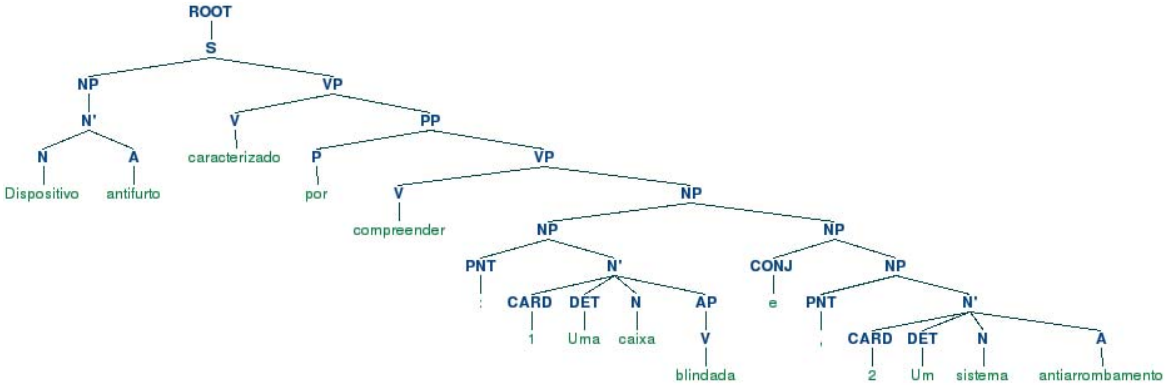

Fig. 2. Syntactic tree for the sentence in Figure 1. Each non-terminal node represents a constituent in the phrase structure decomposition. The leaves are the words from the sentence, with their POS-tags.

sequence for each [segment, syntactic node] pairing, removing the aligned segments from the search. The aligned segments are annotated as class labels in the syntactic tree. Figure 3 shows an annotated syntactic tree.

This type of alignment is based on Noam Chomsky's "structural meaning" notion, further developed by Katz and Fodor [14] in the concept of *projection rules*: mappings between syntactic constituents and their meanings, in the form of semantic markers applied over syntactic elements. Altough parsing is not always precise, the alignment is also robust with respect to parsing errors due to the fact that similar phrase constructions often lead to the same errors.

c) Classifier training: With the attached syntactic information, the segments can be analyzed for a set of structural properties contributing to their semantic function. The properties used in this work are:

- *POS-tag count*: number of times each POS class occurs in the segment;
- *POS-tag position*: appearance order of each POS class in the segment;
- *Word count*: number of words in the segment;
- *Syntactic tag*: syntactic class of the node aligned to the segment;
- *Syntactic parent*: syntactic class of the parent of the node aligned to the segment;
- *Semantic parent*: class of the parent node on the semantic segment tree;
- *Semantic predecessor*: class of the last read segment.

The values obtained for each segment are used as training input for the WiSARD weightless neural network classifier [15]. The values are encoded as binary strings as required by the WiSARD model. The WiSARD classifier is used for determining the class of a segment in the next phase.

d) Filter training: In this step, two types of filter are trained with the annotated segments: a segment hierarchy filter and a syntactic-semantic pair filter. For the first one, the pairs [parent, child] found on the presented semantic segment trees are recorded, and for the second, pairs [segment class, syntactic class] are recorded. Filters are used to eliminate or disambiguate segments of unseen claims that do not fall under the recorded cases.

C. Extraction and annotation phase

The extraction and annotation phase comprises:

1) Extraction of "candidate segments" from unseen claims;
2) Validation and classification of "candidate segments", which will then be annotated with their respective semantic function classes;
3) Extraction of the relations between segments for the knowledge graph construction.

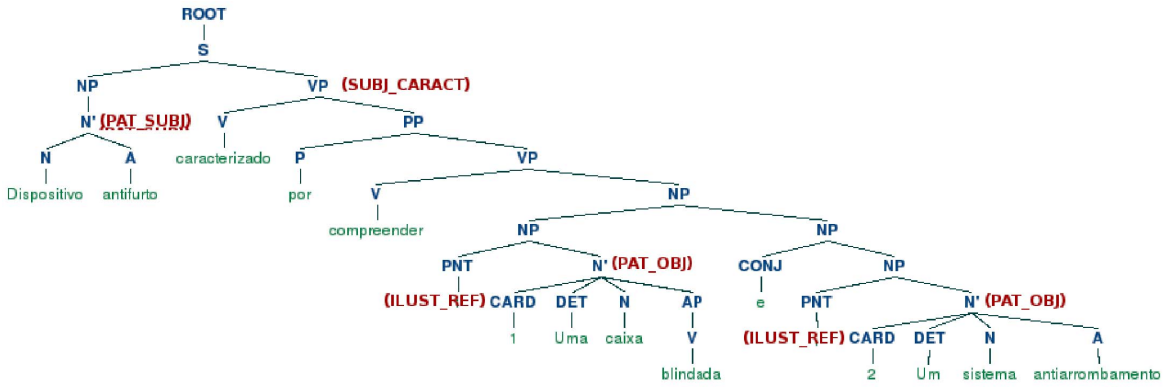

Fig. 3. Syntactic-semantic aligned tree. Syntactic nodes that contain all the words in a semantic segment are annotated with the class of the segment.

In this phase, the unseen claims are presented to the system in plain text. Each claim is processed through the following steps:

a) Parsing: The first step of the extraction phase is the same as the training phase, but with no corresponding semantic segment tree for alignment. Thus the entire syntactic tree is used for the next step.

b) Segment extraction and classification: The syntactic three is walked depth-first, verifying for each node if its tag, i.e. syntactic class, is recorded in the syntactic-semantic pair filter. For the ones that are recorded, values of structural properties are calculated in the same way as in the training phase. These actions consider that there is a semantic segment aligned to the node, except for the semantic parent and semantic predecessor, which are filled by keeping a stack with the last classified segments. The semantic predecessor is the last one classified and is found on the top of the stack. To find the semantic parent, the stack is searched top-down for the last pair [parent, child] occurring in the claim that is also recorded in the segment hierarchy filter. The *parent* element from the pair is selected as the semantic parent (the *child* element is the current node). Since the system does not know the semantic function class of the current node at this point, the selected semantic parent is a guess, and may be revised after the node classification.

Thereafter, the properties are converted into binary strings and passed to the trained WiSARD classifier. With the classifier's response, the segment class pair [parent, current node] is now known by the system and the segment hierarchy filter is applied once more. If the pair is not recorded in the filter, the *bleaching* technique [16] is applied to the WiSARD classifier to reduce a possible overtraining effect, and the classifier is run again. This process is repeated until the segment class pair passes the filter and the segment class is chosen, or until bleaching is not possible and the node is discarded. Successfully classified nodes are annotated with the chosen class and added to the top of the classified segments stack. This step has a similar role to the concept extraction process presented in [4], but aiming to the capture of the trained segment patterns instead of domain terms.

c) Segment relation extraction: In this step, the system does a bottom up search on the classified segments' stack, reading the segments in the order they appear in the sentence and matching the sequence of segment classes to a set of manually coded RegEx[6]-like rules. Each matching yields a relation, for which the meaning is encoded in the ruleset as a label. Matched segments are linked by the relations found, resulting in a semantic relations graph, with segments and relations as nodes and edges respectively. This approach is role-centric, using segment classes and positions for the extraction, similar to the work of Girju et al. [7] for part-whole relations, but applied to generic patterns. It can extract different types of relations from the ones found in the verb-centric approaches, such as [4], or in the document-centric ones, such as [6], with the focus being on the semantic functions described in Section III-A. Figure 4 shows the resulting semantic relation graph for the claim in Figure 1, Table I shows the rules used for the Brazilian patent claims; graph relations are written in RDF n-triples format [17]. A flowchart of the entire system is shown in Figure 5.

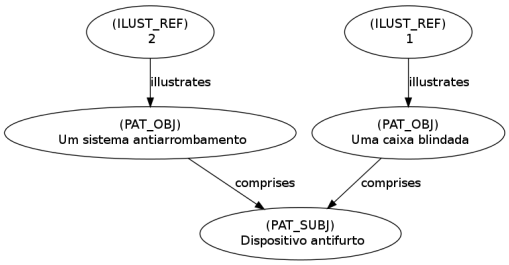

Fig. 4. Semantic relation graph from claim. The graph exposes the semantic structure of the claim. The claim shown in the figure characterizes the claimed subject by describing its components, which are illustrated in the patent document.

D. Experiments

The system was tested using documents from *INPI*, the Brazilian national patent office. Most patent documents pub-

[6]Regular expression

TABLE I. RULESET FOR SEMANTIC RELATION EXTRACTION

Sequence	Subject	Object	Relation (predicate)
PAT_SUBJ, REF, REF_REIVIND	PAT_SUBJ	CLAIM_REF	according to
CLAIM_REF, CLAIM_REF_NUM	CLAIM_REF_NUM	CLAIM_REF	identifies
PAT_SUBJ, *, SUBJ_CHARACT, *, PAT_OBJ	PAT_OBJ	PAT_SUBJ	(verb used in SUBJ_CHARACT)
PAT_OBJ, OBJ_CHARACT, PAT_OBJ	PAT_OBJ	PAT_OBJ	(verb used in OBJ_CHARACT)
PAT_OBJ, ILUST_REF	ILUST_REF	PAT_OBJ	illustrates
PAT_OBJ, OBJ_CHARACT, [^PAT_OBJ][7]	OBJ_CHARACT	PAT_OBJ	Characterizes

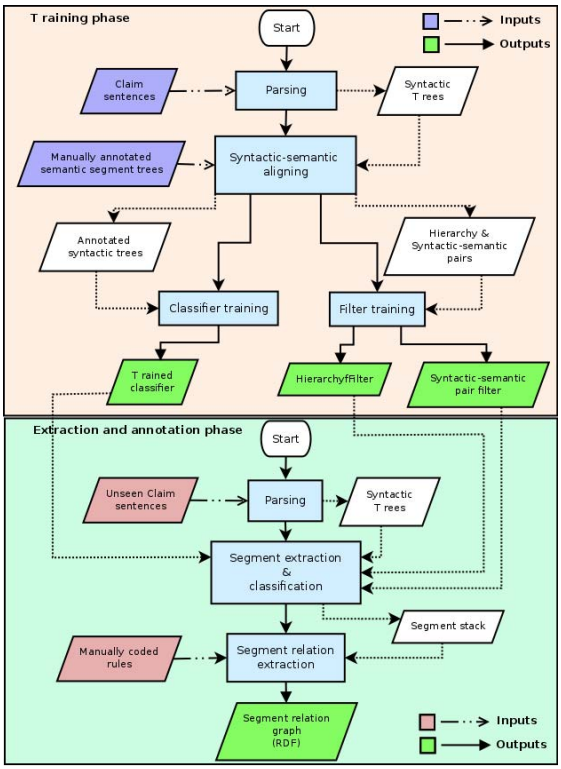

Fig. 5. Flowchart of the semantic extraction and annotation system. Boxes represent the processing steps and rhomboids represent data items: inputs or outputs for each step. External inputs and outputs for each phase are highlighted.

lished by INPI have restricted access; only the summaries are available. The documents fully available to the public are published as scanned paperforms, in *PDF* format. Thus, the only way to access their textual content is by performing *OCR* [8] on the PDF files. In this work, the open source tool Tesseract-OCR [18] was used for this task with good results. However, some manual corrections in the text were still necessary, specially on accents and similar characters like "**l**" and "**1**", "**O**" and "**0**". After obtaining the text, separating the claims from the rest of the document (*spotting*) is also non-trivial, due to the lack of a standard structure for the documents, with many changes in the form type over the years. This work assumed spotting had already been done for the text input, and the text used in the tests was manually spotted. Once claims were isolated, the next difficulty is the syntactic parsing, as parsers for Brazilian Portuguese are still not on par with state-of-art English parsers in terms of accuracy. Initial tests showed that accuracy could be substantially increased in existing Portuguese parsers, by using a more precise POS-tagger. This was possible by using the LX-Parser [19], for syntactic parsing, combined with the mWANN-tagger [20], for POS-tagging. This set of obstacles makes Brazilian patent documents a relevant choice for this type of application. However, due to such difficulties and time constraints, a very small sample of 5 patent documents was used in the experiments. From this sample, a set of 10 claims was randomly selected, from which 123 semantic segments were manually labeled for training and testing the system. Annotation was done by a single person.

Testing was performed in two different stages: classification and extraction. Classification testing was performed by collecting the system training output as a list of properties for each semantic segment. This list was used to train and test another classifier system in order to evaluate the discriminative power, i.e., the ability of the system to correctly classify a segment, given that the segment was already correctly extracted. For this stage, all labeled segments were used in a 10-fold cross-validation test on a multilayer perceptron classifier, with a simple 2-layer configuration. Extraction testing was performed by collecting the final output of the system and comparing it to the ground truth data, i.e. manually annotated segments, to verify if the segments were correctly extracted and classified. This verification was done by means of a string comparison, for the purpose of automation, with a $\geq 75\%$ overlap threshold from the extracted segment to the ground truth segment. Thus, no agreement between annotators was needed. The threshold was selected after careful consideration of the impact in accuracy caused by the inclusion or the exclusion of single words (prepositions, conjunctions and punctuations) in the start and in the end of extracted segments. For this stage, a 2-fold cross validation test was employed using the 10 selected claims, with an average of 61 segments per fold. Due to the use of simple sequence matching rules, the segment relation extraction accuracy is directly tied to the extraction precision, and thus was not evaluated. Table II summarizes the testing results.

Results show that the method succeeds in distinguishing segment classes, but still have difficulty with identifying the segments from the rest of the text. The method correctly separates the claims' segments, so extraction recall is relatively high, but precision is still relatively low due to structural similarities between segments of interest and others. Filters

[8]Optical Character Recognition

TABLE II. TESTING RESULTS (WEIGHTED AVERAGE). *Classification* MEASURES THE SYSTEM PERFORMANCE ON IDENTIFYING THE CORRECT SEMANTIC CLASS OF UNSEEN CORRECTLY EXTRACTED SEGMENTS, WHILE *Extraction* MEASURES THE PERFORMANCE ON BOTH EXTRACTING AND CLASSIFYING SEGMENTS FROM UNSEEN CLAIMS.

Test	Precision	Recall	F-measure
Classification	0.94	0.93	0.93
Extraction	0.42	0.70	0.52

are meant to reduce false positives, but are still insufficient to raise the accuracy substantially.

Experiments did not include a speed benchmark, but some time measurements were made in order to evaluate the cost of each component. They are shown in Table III.

TABLE III. TIME MEASUREMENTS (MEAN TIME PER CLAIM) ON A INTEL® ATOM™ N270 1.6 GHZ SINGLE CPU, 2 GB RAM COMPUTER.

Component (training)	Time (claim)	%
Tokenizing + Tagging	4.5s	42%
Parsing	5.5s	51%
Alignment	0.1s	1%
WiSARD training	0.6s	6%

Component (testing)	Time (claim)	%
Tokenizing + Tagging	6s	46%
Parsing	6s	46%
Classification + Extraction	1s	8%

IV. CONCLUSION

This paper presented a method for extracting semantic information from patents with the use of automatically annotated phrasal structures. This was achieved by using the semantic segment concept as a model for semantic function classification, which enabled syntactic-semantic alignment using syntactic trees and with that, the calculation of structural properties contributing to semantic function. The method abstracted domain ontology information for generality, and output the information in an ontology-friendly format for posterior ontology alignment. The semantic segment extraction system was briefly evaluated with patent documents from *INPI*, the Brazilian patent office. Despite the obstacles in document retrieval and processing, it showed promising results, in particular for the classification capabilities. The extraction results indicated a vast room for improvement in the segment filters, as a way of eliminating false positives. It is important to note that the presented method is general-purpose and does not benefit from corpus specific analysis. Therefore, no comparison with patent specific methods was made and it could be applicable to other corpora. Future work will focus on increasing the number of document samples, integrating with ontology alignment mechanisms, and also adding new information to the segment extraction step, e.g. rules over POS-tag sequences.

ACKNOWLEDGMENT

This work was partially supported by Inovax, and CAPES, CNPq, FAPERJ and FINEP Brazilian research agencies.

REFERENCES

[1] N. Ghoula, K. Khelif, and R. Dieng-Kuntz. *Supporting patent mining by using ontology-based semantic annotations*. In Web intelligence, IEEE/WIC/ACM international conference on (pp. 435-438). IEEE, 2007.

[2] S. Taduri, G. T. Lau, K. H. Law, and J. P. Kesan. *A patent system ontology for facilitating retrieval of patent related information*. In Proceedings of the 6th International Conference on Theory and Practice of Electronic Governance (pp. 146-157). ACM, 2012.

[3] D. Y. Yang, and V. M. Soo. *Extract conceptual graphs from plain texts in patent claims*. Engineering Applications of Artificial Intelligence, 25(4), (pp. 874-887), 2012.

[4] V. H. Ferreira, L. Lopes, R. Vieira, and M. J. Finatto. *Automatic Extraction of Domain Specific Non-taxonomic Relations from Portuguese Corpora*. In Web Intelligence (WI) and Intelligent Agent Technologies (IAT), IEEE/WIC/ACM International Joint Conferences on (Vol. 3, pp. 135-138). IEEE, 2013.

[5] M. Bruckschen, J. G. C. de Souza, R. Vieira, and S. Rigo. *Sistema SeRELeP para o reconhecimento de relações entre entidades mencionadas*. Mota and Santos (Mota and Santos, 2008), 2008.

[6] G. M. Caputo. *Sistema Computacional para o processamento textual de patentes industriais*. Universidade Federal do Rio de Janeiro, 2006

[7] R. Girju, A. Badulescu, and D. Moldovan. *Automatic discovery of part-whole relations*. Computational Linguistics, 32(1), 83-135, 2006.

[8] P. Pantel, and M. Pennacchiotti. *Espresso: Leveraging generic patterns for automatically harvesting semantic relations*. In Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics (pp. 113-120). Association for Computational Linguistics, 2006.

[9] Word Wide Web Consortium (W3C). *Resource Description Framework*. http://www.w3.org/RDF/

[10] D. Moldovan, A. Badulescu, M. Tatu, D. Antohe and R. Girju *Models for the semantic classification of noun phrases*. In Proceedings of the HLT-NAACL Workshop on Computational Lexical Semantics (pp. 60-67). Association for Computational Linguistics, 2004.

[11] Virtuoso Universal Server (Openlink Software). http://virtuoso.openlinksw.com/

[12] Apache Jena - Semantic Web Framework. https://jena.apache.org/

[13] M. Marcus, M. A. Marcinkiewicz, and B. Santorini. *Building a large annotated corpus of English: The Penn Treebank*. Computational linguistics 19.2 (pp. 313-330), 1993.

[14] J. J. Katz, and J. A. Fodor. *The structure of a semantic theory*. Language (pp. 170-210), 1963.

[15] I. Aleksander, W. V. Thomas, and P. A. Bowden. *WISARD· a radical step forward in image recognition*. Sensor review 4.3 (pp. 120-124), 1984.

[16] D. S. Carvalho, H. C. C. Carneiro, F. M. G. França, P. M. V. Lima. *B-bleaching: Agile Overtraining Avoidance in the WiSARD Weightless Neural Classifier*. European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning (pp. 515-520), 2013.

[17] Word Wide Web Consortium (W3C). *RDF N-triples*. http://www.w3.org/TR/n-triples/

[18] R. Smith. *An Overview of the Tesseract OCR Engine*. ICDAR. Vol. 7. (pp. 629-633), 2007.

[19] J. Silva, A. Branco, S. Castro and R. Reis. *Out-of-the-Box Robust Parsing of Portuguese. In Proceedings of the 9th International Conference on the Computational Processing of Portuguese* PROPOR'10, (pp. 75–85), 2010.

[20] H. C. C. Carneiro and F. M. G. França and P. M. V. Lima. *WANN-Tagger: A Weightless Artificial Neural Network Tagger for the Portuguese Language*. Proceedings of ICFC-ICNC 2010, (pp. 330-335), 2010.