A Graph Knowledge-base for Auditing Human Resources Public Management

Fernando Antonio D. G. Pinto[1], **Sérgio Lifschitz**[1], **Edward Hermann Haeusler**[1]

[1]Departamento de Informática – Pontifícia Universidade Católica do Rio de Janeiro (PUC-Rio)
Rio de Janeiro – RJ – Brazil

{fpinto, sergio, hermann}@inf.puc-rio.br

Abstract. *Given the growing volume of data that is generated in the public sector, new payroll audit models are created. Payroll is a sensitive area and requires assertive processes in its operations. The formal verification of public rules, in a knowledge base, aims to demonstrate the conformity of public acts in relation to the respective legal norms. In this paper we build a knowledge base of public acts from the extraction of information contained in a 9-year series of a official gazette. With this, we can audit the participation of public servers in technical committees through SPARQL queries in a knowledge base such as RDF graphs.*

1. Introduction

Public auditing is an important activity that aims to help organizations improve their internal processes. Nowadays, laws that regulate the efficiency of public management, such as payroll expenses, make it necessary to invest in management tools that help in the inspection of public spending. Often, the volume of data is growing, and that public manager, using traditional payroll tools, feels difficult to identify critical points in human resources and payroll systems. Therefore, we present an audit tool approach that uses resources from a shared web, and that is possible for inspection by people or machines, helping the public manager to make his own inferences in a knowledge base.

In this research, the entities extracted were tripled to an RDF format and can be further submitted to *SPARQL* queries in a triple store database, such as the AllegroGraph [Buil-Aranda et al. 2013]. The extraction of the acts from the gazettes to a knowledge base is part of a wider project of KB creation for public documents in the context of e-governance audit and compliance.

This paper is divided into the following sections: Section 1, this introduction that presents our research motivation. Section 2, we introduce some necessary concepts like *RDF*, *iALC* and Regular Expressions (background). Section 3, discusses Related Works. Section 4 presents the challenges and research proposal. In Section 5, we present the Results, and finally, Section 6 we make a brief Conclusion of the research and Future Works.

An important observation is that due to the Brazilian General Data Protection Law (LGPD) rules [Brasil 2018] and [Brasil 2019], some data presented in the images of this work were anonymized.

2. Background

2.1. Resource Description Framework

The Resource Description Framework (RDF) is a data model (metadata) used for information representation in order to make it easier to search for resources on the Web.

In [Group 2014] and [Barzdins et al. 2019] *RDF* increases the structure of Web links when using URIs to give a name to the relationship among resources on the Web, available in the form of a triple <subject>, <predicate> and <object> elements. This model represents a directed labeled graph (RDF Graph) where the endpoints(nods) represent resources (<subject> or <object>) that are related by a predicate (edge) [Isotani and Bittencourt 2015]. These resources can mean any information (document, person, etc.). In this case, each resource is assigned an identifier element Internationalized Resource Identifiers, or IRI. Hence, the RDF model allows structured and semi-structured data to be combined, exposed and shared across different technologies on the Web.

Roughly speaking, in addition to establishing a model for encoding and transmitting metadata, RDF's main objective is to maximize the interoperability of data from heterogeneous sources on the WEB. This feature justifies our adoption in this project. We want it to be possible to represent and extend a knowledge base on the web.

In this work, we choose to build a knowledge base in the form of RDF triples, as presented in [Amato et al. 2008] and [Najmi et al. 2016]. The Section 4 shows how this was done.

2.2. Intuitionist Description Logic - *iALC*

Description Logic (DL) is a formalism used for knowledge representations. Essentially, DL has three components: Individuals (constants) that represent entities in a domain; Concepts (unary predicates) that are properties given to Individuals; Roles (binary predicates) which are the relationships between Individuals [Baader et al. 2008].

A piece of the alphabet of a DL consists of:

- Set of names of Individuals, Concepts and Roles.
- $\sqcup$ represents the conjunction of Concepts.
- $\sqcap$ represents the disjunction of Concepts.
- $\sqsubseteq$ represents the inclusion of Concepts. The operation $C \sqsubseteq D$ indicates that Concept C is included in Concept D.
- $\forall$ represents the universal restriction of Concepts. Thus, $\forall R.C$ represents the universal restriction of the Concept C under Role R.
- $\exists$ represents the existential restriction of Concepts using Roles.
- $\neg$ represents the complement of Concepts.
- : an assertion operator of type $a : C$ (Concept assertions) and $(a, b) : R$ (Rules assertions). Where C (Concept), R (Role), a and b (Individuals), these operations indicate that the Individuals apply the Concept or Role to which they refer.

The iALC logic, derived from ALC, is an intuitionist description logic created to deal with legal texts. Section 6 presents a formalization using logic iALC for a particular case.

2.3. Regular Expressions

Regular Expression (RE) is a notation for specifying lexeme patterns. Its syntactic construction is composed of atomic symbols (characters), union, concatenation and closing of *Kleene* of other regular expressions. Readers unfamiliar with the concept and terminology related to regular expressions can refer to the book [Aho et al. 2006].

3. Related Works

During this research, three articles were found that present techniques of information retrieval in the Official Gazette.

The first [Rodríguez and Bezerra 2019] uses Natural Language Processing techniques [Friedman et al. 2013] to recognize Named Entities in the appointment ordinance on the Official Gazette. They use the resources available on the Natural Language Toolkit (NLTK) platform for steps of the tokenization process until the entity recognition. A limitation of this work is that the authors present a tool that recognizes only the Names of public agents (public servants) in the appointment ordinance. In this experiment, it was possible to observe an accuracy of 92% in the extraction of names.

The second [Junior et al. 2018] uses Data Mining techniques for information retrieval in the Official Gazette of the Government of Pernambuco/Brazil. That research applied the Random Tree algorithm with a hit rate of 80%. The authors agree that "if the department wants an algorithm with better results, it is necessary to carry out a minimum standardization of the Official Gazette so that the extraction is more efficient". This highlights the complexity in the treatment of data contained in the official gazette. In this case, a study of other information retrieval strategies is necessary.

Finally, in the third paper [Pinto et al. 2021] presents a proposal for extracting these public data by regular expression techniques. As a result, it promotes public transparency and possibly aids government decision-making process by building a public knowledge base driven by the grammar and entities of the Official Gazettes. The scope was limited to publications involving personnel movements in the last nine years.The result was a Knowledge Base RDF of 24,745 public acts for *SPARQL* queries.

4. Research Design and Methods

4.1. Scope

The first challenge was to capture official gazettes for a period. For this activity, and others of this project, the language Python, version 3, was used as a *backend* of the production tool and generation of the RDF triples of public acts published in the Official Gazettes. In order to highlight the contributions, we will restrict ourselves here to the City of Rio de Janeiro[1]. The scope was restricted to publications involving the creation of administrative committees and their members.

As presented, the Python module was developed, as seen in Fig. 2 in order to automate the download of the official gazettes used in the project. In total, 2,949 official gazettes were recovered and treated (since 2013). With these documents, the second step was to build the extractor of information contained in each official gazette. In this process,

[1]https://doweb.rio.rj.gov.br/

we used the *RE*[2] library from *Python* and a set of regular expression patterns to compose a set of rules based on grammars (Listing 1) of the acts published.

To exemplify the use of our tool, we decided to treat a subset of human resources and administrative information involving the publications that create administrative committees and their members.

Based on this scope, pattern extractors were developed using regular expression techniques applied to the grammar of the acts targeted in this research. Naturally, regular expression algorithms tend to be greedy and identifying the grammar of posts and mapping them to their corresponding regular expressions made the tool quite efficient.

4.2. Official Gazette

According to Article 37 of the Constitution, Brazilian public administration is supported by 5 principles:

Legality - all public acts must be guided by the law. **Impersonality** - personal interests cannot override the public interest, and state power cannot be used for personal gain either. **Morality** - Public servants and other State workers must follow ethical and moral standards. **Publicity** - all acts of public administration must be done with the knowledge of the population, that is, they must be publicized so that everyone is aware. Public documents need to be accessible to everyone. **Efficiency** - the service provided by the public administration needs to be efficient, have the best result at the lowest possible cost and in the fastest way, always aiming at quality.

According to the principle of publicity, it seems reasonable that the Official Gazette becomes one of the primary sources of information for the public administration and its extraction becomes almost a necessity. Fig. 1 shows examples of public acts that create a technical committee.

The first challenge was to capture official gazettes for a period between 2013 and 2022. For this activity, and others of this project, the language Python, version 3, was used as a backend of the production tool and generation of the RDF triples of public acts published in the Official Gazettes. In order to highlight the contributions, we will restrict ourselves here to the City of Rio de Janeiro. However, the results may easily extend to any other city one may investigate with data publicly available. For instance, We have already extracted data on human resources from the cities of Maceió, Recife, Santa Catarina and Palmas.

Initially, we have developed a *script* to *download* the files considered in the project with the *requests* method of *Python*. A total of 2,949 official gazettes were automatically processed.

We could identify about two patterns in regular expressions in those available public acts. The result of executing our algorithm was the extraction of information contained in 35,014 public acts.

4.3. Grammar and Patterns of the Official Gazette

Parsing Expression Grammars (PEG) is a formalism that describes language recognizers and is a simpler alternative to presenting the syntactic formation rule (grammar) of

[2]https://docs.python.org/3/library/re.html

ATO DO SECRETÁRIO
RESOLUÇÃO SMS Nº 2587 DE 14 DE ABRIL DE 2015

Designa os membros da Comissão Técnica de Acompanhamento (CTA) do Contrato de Gestão nº 006/2011 referente ao processo instrutivo nº 09/000.008/2015 (Programa Saúde na Escola – PSE).

O SECRETÁRIO MUNICIPAL DE SAÚDE, no uso de suas atribuições que lhe são conferidas pela legislação em vigor.

CONSIDERANDO §2º do Artigo 8º da Lei Municipal nº 5026 de 19 de maio de 2009 que prevê a análise dos resultados atingidos com a execução do contrato de gestão por Comissão de Avaliação;

CONSIDERANDO a necessidade de monitoramento e avaliação das ações e serviços de saúde prestados pelas Organizações Sociais de Saúde que possuem contrato de gestão com a Secretaria Municipal de Saúde;

CONSIDERANDO a necessidade da revisão da composição da Comissão Técnica de Acompanhamento (CTA) do Contrato de Gestão.

RESOLVE:
Art. 1º Designar os membros abaixo indicados para comporem a Comissão Técnica de Acompanhamento:

COMPOSIÇÃO CTA – PSE		
TITULARES		
ÓRGÃO	**NOME**	**MATRÍCULA**
S/GAB	RODRIGO DO SOUSA PRADO	11/229.220-9
S/SUBPAV	LUCIANA SOARES RIBEIRO	11/227.277-1
S/ACS	CLAUDIA DE OLIVEIRA FARIA FERRARI QUADROS	11/157.497-9
S/SUBG	ANTONIO RICARDO GOMES JUNIOR	60/274.497-7
S/SUBG	MICHELLE RODRIGUES SCHINKE	11/226.675-7
SUPLENTES		
ÓRGÃO	**NOME**	**MATRÍCULA**
S/SUBPAV	NINA LUCIA PRATES NIELEBOCK	57/160.631-8
S/SUBG	ANA CAROLINA HENRIQUE SIQUEIRA LARA	60/262.710-7

Art. 2º Esta Resolução entra em vigor em **01 de agosto de 2014.**
Rio de Janeiro, 14 de abril de 2015.
DANIEL SORANZ

Figure 1. Example of a public acts published in the Official Gazettes.

```
⟨publicAct⟩  ::=  ⟨top⟩⟨segment⟩
      ⟨top⟩  ::=  RESOLUÇÂO SMS No.⟨port⟩ DE ⟨per⟩
      ⟨per⟩  ::=  ⟨day⟩ DE ⟨month⟩ DE ⟨year⟩
  ⟨segment⟩  ::=  ⟨segment1⟩⟨segment2⟩
 ⟨segment1⟩  ::=  Designa os membros da ⟨cta⟩ do Contrato de Gestão n°⟨contrato⟩
 ⟨segment2⟩  ::=  referente ao processo instrutivo n°⟨numProcesso⟩⟨descContrato⟩
      ⟨cta⟩  ::=  Comissão Técnica de Acomapanhamento⟨tpoComissao⟩
     ⟨port⟩  ::=  [0-9]+
      ⟨day⟩  ::=  [0-9]+
    ⟨month⟩  ::=  [A-Z]+
     ⟨year⟩  ::=  [0-9]+
 ⟨contrato⟩  ::=  [0-9/]+
⟨numProcesso⟩ ::=  [0-9/.]+
⟨descContrato⟩ ::=  [A-Z0-9/.-]+
⟨tpoComissao⟩ ::=  (CTA)
```

Listing 1. Piece of Official Gazette grammar.

certain languages. Here, we present an example of the PEG, Listing 1, identified in the Official Gazette. In this case, this piece of grammar corresponds to the publication header presented in Fig. 1.

From the PEGs, it is possible to perform a direct translation into regular expressions that represent the rules for extracting information. For reasons of space we prefer to omit these examples, but they can be easily seen in the project repository [3]. We believe that this formalism, present in the official gazette, makes this research quite reasonable.

4.4. The Encoding Process

With the information extracted, in compliance with regular expression standards, we started the process of triplification of the information to an RDF format. For this step, the RDFLIb [Team 2013] library for Python was used. The library has interfaces that simplify and facilitate the implementation of RDF nodes. Optionally includes parsers for RDF/XML, N3, NTriples, N-Quads, Turtle, TriX, RDFa, and Microdata. It implements a Graph interface in which we can store graph information in memory or persistent storage. It is also possible to run queries and updates in the SPARQL language. In this work, we chose to serialize the data in RDF/XML format.

An essential step in the project was the ontology to give meaning to the contents of the Official Gazette. In such a case, as a basic proof of concept, we chose to define an adapted and generic ontology (Listing 1) in *"Friend of a Friend" (FOAF)* [Brickley and Miller 2014]. This ontology enabled queries in our knowledge base stored in *AllegroGraph* [AllegroGraph 2019].

[3]https://github.com/fernandoantoniodantas/COMISSOES2RDF

```
for row in funcionarios_records:
    seqa+=1
    idP = seqa
    idP = BNode()
    store.add((idP, RDF.type, FOAF.Comissoes))
    store.add((idP, FOAF.tpoComissao, Literal(row[0].strip())))
    store.add((idP, FOAF.portaria, Literal(row[1])))
    store.add((idP, FOAF.dataPub, Literal(row[2])))
    store.add((idP, FOAF.nomecomissao, Literal(row[3].strip())))
    store.add((idP, FOAF.orgao, Literal(row[4])))
    store.add((idP, FOAF.matricula, Literal(row[5].strip())))
    store.add((idP, FOAF.numContrato, Literal(row[6].strip())))
    store.add((idP, FOAF.descContrato, Literal(row[7].strip())))
    store.add((idP, FOAF.numProcesso, Literal(row[8].strip())))
    store.add((idP, FOAF.nome, Literal(row[9].strip())))
    # Serialize the store as RDF/XML to the file DO2RDF.rdf.
    store.serialize("RDF/DO2RDF_COMISSOES.rdf", format="pretty-xml",
    max_depth=3)
    print('RDF Serializations:', seqa, 'De', size)
```

Code Listing 1. RDF/XML serialization with adapted ontology.

The Listing 2 is a piece of file COMISSAO2RDF.rdf, whih shows the result serialization process with RDFLib. This file is used to deploy in AllegroGraph.

```
<?xml version="1.0" encoding="utf-8"?>
<rdf:RDF
  xmlns:foaf="http://xmlns.com/foaf/0.1/"
  xmlns:rdf="http://www.w3.org/1999/02/
  22-rdf-syntax-ns#">
  <foaf:Comissoes rdf:nodeID="N7654fa3584ea89c0ade">
    <foaf:nome>FÁTIMA CRISTINA CUNHA PENSO</foaf:nome>
    <foaf:matricula>10/209.246-8</foaf:matricula>
    <foaf:portaria>2162</foaf:portaria>
    <foaf:dataPub>2013-08-12</foaf:dataPub>
    <foaf:secretaria>S/SUBHUE</foaf:secretaria>
    <foaf:numProcesso>09/003752/11</foaf:numProcesso>
    <foaf:tpoComissao>CTA</foaf:tpoComissao>
    <foaf:contrato>003/2012</foaf:contrato>
    <foaf:descContrato>Maternidade Zona Oeste</foaf:descContrato>
  </foaf:Comissoes>
```

Listing 2. Serialization in RDF/XML.

4.5. Architecture

Fig. 2 shows the high-level architecture of our extraction and audit tool.

The modules are represented as follows:

- **Crawling Module** - It is responsible for retrieving the official gazettes files automatically.
- **PDF File** - Set of Official Gazette in PDF format.
- **Regular Expression Patterns** - Regular expressions used to information extraction.

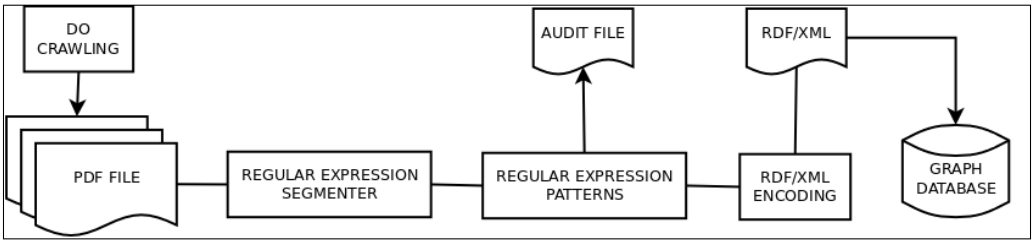

Figure 2. DO2RDF tool architecture.

- **Regular Expression Segmenter** - Module for partitioning the Official Gazette, separating the target publications from the other publications. Providing the Regular Expression Patterns module with a segment with the detected grammar.
- **RDF/XML Encoding** - Module that transforms the extracted information into RDF triples.
- **RDF/XML** - Serialization file with RDF triples.
- **Graph Database** - Graph database for *SPARQL* queries containing RDF triples.

5. Results

We present in this section the results obtained after extracting the information contained in the gazettes and the triplifying step in *RDF/XML* data to be executed in *SPARQL* query language environment.

With the extracting done, we started the information tripling process. For this step, we used the library *RDFLib* [Team 2013] for *Python*. The library has interfaces that make it simple and easy to implement *RDF* nodes. Optionally, it includes parsers and serializers for *RDF/XML*, *N3*, *NTriples*, *N-Quads*, *Turtle*, *TriX*, *RDFa* and *Microdata*. It implements a *Graph* interface to which we can store graph information either in memory or persistent storage. It is also possible to run queries and updates in the *SPARQL* language.

We have chosen in this work to serialize the data in the *RDF/XML* format for loading in a query environment. In this case, *AllegroGraph* [AllegroGraph 2019] was used. An important step was the choice of the ontology to be used for the definition of meanings regarding the contents of the Official Gazette. Initially, as a basic proof of concept, we chose to define an adapted and generic ontology based on *"Friend of a Friend" (FOAF)* [Brickley and Miller 2014]. The Listing 1 presents the *kernel* of the serialization module in *RDF* of the official gazette data.

To execute the *SPARQL* queries, we considered *AllegroGraph* version 6.6.0 [AllegroGraph 2019] running in a virtual environment with Ubuntu GNU/Linux. Subsequently, the repository was created to receive the load of the file *DO2RDF_-COMISSOES.rdf*. Recall that the choice for a knowledge base in RDF was already justified in the subsection 2.1.

Just to illustrate the use of our approach, we executed some *SPARQL* queries in this RDF graph environment.

The Listing 3, a simple *SPARQL* query that counts and groups information by employee registration number. Our objective is to identify the employees with more than 1 participation in public committee in descending order.

```
SELECT (COUNT(?Matricula) as ?count) ?Matricula ?Nome
where{
?person foaf:nome ?Nome .
?person foaf:matricula ?Matricula .

} group by ?Matricula ?Nome HAVING (?count > 1)
order by DESC(?count)
```

Listing 3. *SPARQL* queries for committee history.

As shown in this Figure(3), the result of this *SPARQL* query retrieves some interesting cases for analysis. In the first line, employee Marcos [...] Santos participated in 151 committees. For more details about this case, the *SPARQL* query, Listing 4, retrieves some data: date of publication in the official gazette, ordinance, process number, the contract and its description for this employee, Figure 4.

count	Matricula	Nome
"151"	"11 -6"	"MARCOS SANTOS"
"57"	"11 2"	"FERNANDO
"39"	"11 -0"	"RAPHAEL
"37"	"11 -2"	"CARLA
"35"	"11 -3"	"CONRADO
"31"	"11 -2"	"CRISTINA
"25"	"11 -5"	"HUGO
"22"	"10 -8"	"KATIA
"20"	"11 4"	"ANA
"19"	"10 7"	"IVN
"19"	"11 -0"	"NICOLE
"18"	"11 -9"	CARLOS"
"18"	"11 -1"	"MARIA
"18"	"1 -8"	"MARCIO
"16"	"11 -7"	"MICHELLE

Figure 3. Number of participations in committees per employee.

```
SELECT ?Data_Publica ?Portaria ?Numero_Processo
?Numero_Contrato ?Descricao_Contrato ?Matricula ?Nome
where{
?person foaf:nome ?Nome .
?person foaf:matricula ?Matricula .
?person foaf:dataPub ?Data_Publica .
?person foaf:portaria ?Portaria .
?person foaf:numContrato ?Numero_Contrato .
?person foaf:numProcesso ?Numero_Processo .
?person foaf:descContrato ?Descricao_Contrato .
FILTER (?Matricula='11/131.404-6') .
  } order by ASC(?Data_Publica)
```

Listing 4. SPARQL query for a specific employee.

Finally, our last query (Listing 5) aims to retrieve information relating to employees and their participation in committees, as well the period in which they participate or participated in these committees.

Data_Publica	Portaria	Numero_Processo	Numero_Contrato	Descricao_Contrato	Matricula	Nome
"2013-05-09"	"2067"	"09/004.994/09"	"005/2009"	"CAP-3.1"	"11 6"	"MARCOS SANTOS"
"2013-05-09"	"2066"	"09/004.993/09"	"006/2009"	"CAP-2.1"	"11 6"	"MARCOS SANTOS"
"2013-05-09"	"2065"	"09/002.080/11"	"008/2011"	"CAP-1"	"11 6"	"MARCOS SANTOS"
"2013-05-10"	"2071"	"09/005.019/09"	"004/2009"	"CAP-3.3"	"11 6"	"MARCOS SANTOS"
"2013-05-10"	"2070"	"0000"	"0000"	"0000"	"11 6"	"MARCOS SANTOS"
"2013-05-10"	"2074"	"0000"	"0000"	"0000"	"11 6"	"MARCOS SANTOS"
"2013-05-10"	"2075"	"0000"	"0000"	"0000"	"11 6"	"MARCOS SANTOS"
"2013-05-10"	"2073"	"0000"	"0000"	"0000"	"11 6"	"MARCOS SANTOS"
"2013-05-10"	"2072"	"09/004.435/10"	"004/2011"	"CAP-4"	"11 6"	"MARCOS SANTOS"
"2013-05-10"	"2068"	"0000"	"0000"	"0000"	"11 6"	"MARCOS SANTOS"
"2013-05-29"	"2084"	"09/004.435/10"	"004/2011"	"CAP-4"	"11 6"	"MARCOS SANTOS"
"2013-05-29"	"2085"	"09/004.436/10"	"002/2011"	"CAP-5.1"	"11 6"	"MARCOS SANTOS"
"2013-05-29"	"2087"	"09/004.603/09"	"001/2009"	"CAP 5.3"	"1 6"	"MARCOS SANTOS"
"2013-05-29"	"2086"	"09/004.437/10"	"003/2011"	"CAP-5.2"	"11 6"	"MARCOS SANTOS"
"2013-05-29"	"2082"	"09/004.991/09"	"020/2010"	"CAP 3.2"	"11 6"	"MARCOS SANTOS"

Figure 4. Piece of history of participation in committees.

```
select (COUNT(?Matricula) as ?count) ?Matricula ?Nome
(min(?Data_Publica) AS ?min) (max(?Data_Publica) AS ?max)
(year(?max)-year(?min)AS ?anos)
where {
  ?person foaf:nome ?Nome .
  ?person foaf:dataPub ?Data_Publica .
  ?person foaf:matricula ?Matricula .
} group by ?Matricula ?Nome ?idade HAVING (?count > 1)
order by DESC(?count)
```

Listing 5. SPARQL queries for time in the committees.

The results of this query is shown in Figure 5 where the employee Marcos [...] Santos participate or participated for eight years in committees.

count	Matricula	Nome	min	max	anos
"151"	"11 6"	"MARCOS SANTOS"	"2013-05-09"	"2021-08-10"	"8"
"57"	"11 2"	"FERNANDO"	"2015-06-12"	"2019-06-05"	"4"
"39"	"11 0"	"RAPHAEL"	"2014-09-26"	"2018-05-30"	"4"
"37"	"11 2"	"CARLA"	"2013-08-12"	"2020-09-17"	"7"
"35"	"11 3"	"CONRADO"	"2014-09-26"	"2019-06-05"	"5"
"31"	"11 2"	"CRISTINA"	"2015-12-30"	"2019-07-17"	"4"
"25"	"11 5"	"HUGO"	"2014-09-26"	"2018-05-29"	"4"
"22"	"1 8"	"SILVA"	"2014-09-30"	"2016-04-28"	"2"
"20"	"11 4"	"ANA"	"2016-09-15"	"2019-05-24"	"3"
"19"	"1 7"	"E"	"2015-06-12"	"2016-05-30"	"1"
"19"	"1 0"	"NICOLE"	"2014-02-17"	"2018-11-28"	"4"
"18"	"1 1"	"MARIA OS"	"2013-05-29"	"2019-01-14"	"6"
"18"	"1 8"	"MARCIO"	"2018-04-04"	"2018-06-15"	"0"
"18"	"11 9"	"CARLOS"	"2018-04-04"	"2018-06-15"	"0"
"16"	"11 7"	"MICHELLE"	"2018-04-04"	"2018-05-29"	"0"

Figure 5. Piece of history of participation in committees per year.

This type of audit must still be validated by the competent sector. We noticed during the execution of our experiments that some public acts were published due to inconsistencies. We did not treat these few cases and will leave them as future improvements.

6. Conclusion and Future Works

We presented how extracting information from PDF documents can help in the process of continuous auditing of public acts available in the Official Gazettes. Due to the characteristics of the Official Gazette, the use of regular expressions was presented as a simple and efficient solution. One of the challenges of implementing was the use of the *RDFLib*. This lib showed little flexibility for defining new ontologies. Initially, we tried to modify its *scripts* to incorporate new concepts. Nevertheless, the solution was to adapt the FOAF ontologies to the characteristics of this research. After that, serialization in RDF/XML format became efficient for research purposes and *SPARQL* queries were performed to demonstrate the reasonableness of the tool. Another point to be explored is the definition of a good ontology for the official gazette. As shown, a generic ontology model was used for the purpose of executing the proposal. For an end activity, it is necessary to define the meaning of data present in the RDF model. This will facilitate the process of capturing and processing the data stored in the RDF triples, by machines or users.

Thus, the tool will help public management to infer conditions of non-compliance with the legislation. In the usual way, this example of commission data could help the manager to question whether commissions are being delegated to people on merit or for some other situation of non-compliance with management.

In addition, this research is part of an initiative to analyze and validate the legal norms of governmental acts and decisions. For instance, the definition of an intuitionist version of Description Logic (*iALC*), shown in the previous section 2.2. Thus, maybe thought that future implementation of a reasoner will constructively generate a knowledge base of laws in this language. As an example, we present the modeling of a small part of the legal norm of the City Hall of Rio de Janeiro, referring to the maintenance of the technical committee. The Fig. 6 is an example of this inference process in the knowledge base. In this case, we present a proof using the deductive system for *iALC* that the public employee João cannot participate in a committee for more than two years.

$$\dfrac{\dfrac{joao : Committee \qquad Committee \sqsubseteq \neg moreThan2Years}{joao : \neg moreThan2Years} \qquad [joao : moreThan2Years]}{\dfrac{\bot}{\neg(joao : moreThan2Years)}}$$

Figure 6. Demonstration of the impossibility of participating in committee for more than 2 years.

With this, in the future, a machine will be able to automatically infer on a knowledge base and detect any non-compliance with the legal norm.

References

Aho, A. V., Lam, M. S., Sethi, R., and Ullman, J. D. (2006). *Compilers: Principles, Techniques, and Tools (2nd Edition)*. Addison-Wesley Longman Publishing Co., Inc., USA.

AllegroGraph (2019). Allegrograph - The Enterprise Knowledge Graph. `https://allegrograph.com`. Acessado: 11-12-2019.

Amato, F., Mazzeo, A., Penta, A., and Picariello, A. (2008). Building rdf ontologies from semi-structured legal documents. In *2008 International Conference on Complex, Intelligent and Software Intensive Systems*, pages 997–1002.

Baader, F., Horrocks, I., and Sattler, U. (2008). Chapter 3 description logics. In van Harmelen, F., Lifschitz, V., and Porter, B., editors, *Handbook of Knowledge Representation*, volume 3 of *Foundations of Artificial Intelligence*, pages 135–179. Elsevier.

Barzdins, G., Gosko, D., Barzdins, P. F., Lavrinovics, U., Bernans, G., and Celms, E. (2019). Rdf* graph database as interlingua for the textworld challenge. In *2019 IEEE Conference on Games (CoG)*, pages 1–2.

Brasil (2018). Lei nº. 13.709. Lei Geral de Proteção de Dados pessoais. *Diário Oficial [da] República Federativa do Brasil*.

Brasil (2019). Lei nº. 13.853. Lei Geral de Proteção de Dados pessoais. *Diário Oficial [da] República Federativa do Brasil*.

Brickley, D. and Miller, L. (2014). FOAF Vocabulary Specification 0.99. `http://www.foaf-project.org/`. Acessado: 03-12-2019.

Buil-Aranda, C., Hogan, A., Umbrich, J., and Vandenbussche, P.-Y. (2013). Sparql web-querying infrastructure: Ready for action? In Alani, H., Kagal, L., Fokoue, A., Groth, P., Biemann, C., Parreira, J. X., Aroyo, L., Noy, N., Welty, C., and Janowicz, K., editors, *The Semantic Web – ISWC 2013*, pages 277–293, Berlin, Heidelberg. Springer Berlin Heidelberg.

Friedman, C., Rindflesch, T. C., and Corn, M. (2013). Natural language processing: State of the art and prospects for significant progress, a workshop sponsored by the national library of medicine. *Journal of Biomedical Informatics*, 46(5):765–773.

Group, R. W. (2014). Resource Description Framework (RDF). `https://www.w3.org/RDF/`. Acessado: 11-10-2019.

Isotani, S. and Bittencourt, I. (2015). *Dados Abertos Conectados: em Busca da Web do Conhecimento*. Novatec.

Junior, R., Melo, W., Fagundes, R., and Maciel, A. (2018). Extração de informação e mineração de dados no diário oficial de pernambuco. *Revista de Engenharia e Pesquisa Aplicada*, 3.

Najmi, E., Malik, Z., Hashmi, K., and Rezgui, A. (2016). Conceptrdf: An rdf presentation of conceptnet knowledge base. In *2016 7th International Conference on Information and Communication Systems (ICICS)*, pages 145–150.

Pinto, F. A., Haeusler, E., and Lifschitz, S. (2021). Transparência pública automatizada a partir da gramática do diário oficial. In *Anais do IX Workshop de Computação Aplicada em Governo Eletrônico*, pages 59–70, Porto Alegre, RS, Brasil. SBC.

Rodríguez and Bezerra (2019). Processamento de linguagem natural para reconhecimento de entidades nomeadas em textos jurídicos de atos administrativos (portarias). *Revista de Engenharia e Pesquisa Aplicada*, 5(1):67–77.

Team, R. (2013). RDFLib. `https://rdflib.readthedocs.io/en/stable/#`. Acessado: 13-11-2019.