Knowledge and Information Systems
https://doi.org/10.1007/s10115-024-02077-8

REGULAR PAPER

A semantic-based methodology for the management of document workflows in e-government: a case study for judicial processes

Beniamino Di Martino[1,2,3,6] · Luigi Colucci Cante[1] · Mariangela Graziano[1] · Salvatore D'Angelo[1,6] · Antonio Esposito[1,6] · Pietro Lupi[4] · Rosario Ammendolia[5]

Received: 21 July 2022 / Revised: 27 January 2023 / Accepted: 8 February 2024

Abstract
Trial excessive duration is a common problem in Juridical systems worldwide, even if some countries seem to be more affected by it than others. The European Council has provided metrics and statistics to identify this problem and has pointed out solutions, such as the simplification of norms and the digitization of Juridical procedures. The Italian Telematic Civil Process (TCP) is an example of this digitization effort that has surely positively influenced the duration of Trials, their traceability and general complexity. However, there are still many possible actions that can be taken to simplify the work of Judges and Chancellors, and to support their daily operations in dealing with several Trials at once, and with the consistent number of documents that are involved in them. This paper presents a toolchain and a related methodology for the management of documentation attached to Trials, based on semantic technologies and Natural Language Processing techniques, which will help Judges in faster assessing the situation of each Trial they follow, and will also provide the means to identify potential correlations among different Juridical procedures. The methodology is tested against a case study, i.e. the compensation requests related to road accidents, which has been provided and described by Domain Experts from the Italian Ministry of Justice.

Keywords Semantics · Natural language processing · Name entity recognition · Expert system · BPMN · Trials

1 Introduction

The excessive duration of Trials is a problem that the European Community has addressed several times, trying to suggest both metrics [1] to evaluate the possible causes of such delays,

Luigi Colucci Cante, Mariangela Graziano, Salvatore D'Angelo, Antonio Esposito, Pietro Lupi and Rosario Ammendolia have contributed equally to this work.

✉ Beniamino Di Martino
beniamino.dimartino@unicampania.it

Extended author information available on the last page of the article

and the actions to take to solve the problem [2, 3]. Digitization of the Juridical procedures is one important step toward the solution of the delays in Trials, as it provides the means to constantly monitor the state of such Trials, to track all the actions taken by involved parties easily, and to retrieve the attached documentation.

However, since the amount of documentation that is attached to each Trial can be quite consistent, and despite the strong push toward digitization such documents still lack proper management applications, Judges and Chancellors often lose most of their time in trying to identify all the elements, entities and correlations existing among different dossiers. The Italian Telematic Civil Process (TCP) has, in recent years, brought many advantages to Judges, Chancellors, and Parties, thanks to the simplification of procedures, but it lacks an advanced document management system that can efficiently and efficaciously support Judges and Chancellors in their day to day activities. In particular, many trials need the submission of specific documentation from involved parties, within a limited time frame, otherwise, the Trial is simply dismissed and no further actions are pursued by the Judges. One interesting example is represented by Road Accidents, where involved parties are requested to submit documentation regarding all the events related to the accident, to decide who is going to be compensated by whom, according to the demonstrated responsibilities and damages.

It can be quite difficult for Judges to analyze the huge amount of documentation that such trials generally involve, and it is even cumbersome for them to understand what kind of document has been supplied by the Parties and what is still missing.

The work described in this paper involves the proposal of a toolchain, and a related methodology, for the realization of an integrated system that can be of support to Judges and Chancellors. In particular, the tool chain will support the operations of verification of the presence of all the necessary documentation required by the law that regulates the issue and compliance with delivery times, to pursue a procedure such as a request for damages or invitation to assisted negotiation, representing our reference case studies. A preliminary version of this methodology has already been provided in work [4]. The methodology will apply Natural Language Processing (NLP) techniques to the analysis of the presented documents, or dossiers, to identify the involved entities and their attributes and relationships, and will exploit semantic technologies to build a reliable and robust shared vocabulary, that can help Judges in identify correlations between documents and trials.

The remainder of this paper is organized as follows: Sect. 2 analyzes the current state of the art; Sect. 3 describes the main components of the toolchain and the overall methodology workflow proposed by this paper; Sect. 4 describes the case study of request for compensation in road accidents; Sects. 5.4 and 5.5 present the main NLP techniques used to analyze the dossiers and the generic pipeline that is used to populate the ontologies described in Sect. 5 and report the results obtained with the application of the NLP and Named Entity Recognition (NER) techniques over the analyzed documents; Sect. 5.6 describes the inference rules that will be implemented in the support expert system; finally, Sect. 8 closes the paper with final remarks.

2 Related works

The methodology proposed in this article for the implementation of a tool to support the work of the Judge involves the definition of several activities that must be carried out to implement the proposed complex system. The main activities include: (i) the classification of documents, (ii) the identification of specific entities within documents belonging to a domain

(in our case it is the legal domain), (iii) the semantic annotation of business processes (as well as the realization of a specific business process for the modeled case), (iv) the population and enrichment of ontologies with the results of information retrieval activities, (v) text retrieval through concepts of a domain ontology, (vi) document navigation from business process activities, (vii) the realization of an expert system capable of performing logical inferences on a knowledge base. In order to define a methodology that would allow the implementation of all these activities to be carried out, a series of searches were carried out in the various works already available in the literature, to have a comparison with what has already been done, including in other sectors, and to study the different techniques and strategies applied to define and successively implement a methodology that will be as comprehensive and effective as possible. The proposed methodology includes preliminary activities of document classification and identification of entities belonging to a specific domain within the documents. Over time, more and more progress is being made in this type of activity and the applications of NLP, machine learning (ML), and deep learning techniques are becoming more and more refined and showing their effectiveness in classifying documents and identifying entities in texts. Such activities have been carried out in various domains, and although a number of generic entities, such as the recognition of standard personal data in texts or names of people, places, and organizations, are common in various domains, it is necessary to also have detailed knowledge of terms, concepts, and entities of interest to a particular domain, which in our case is the legal domain. For this reason, in addition to analyzing works with generic applications, we have focused on works that lead the above-mentioned activities to the legal field.

In Martino et al. [5], a framework for building specific test sets to train a named entity recognition model to recognize specific entities in legal texts is presented, while in [6] the authors describe NLP techniques applied to text preprocessing of tweets from the Twitter social network to prepare the test set for training a classification model for the recognition of specific categories of tweets. The classifier proposed in this work is implemented using the Logistic Regressor algorithm [7] offered by the machine learning library of the Big Data platform Spark [8].

In Goncalves and Quaresma [9], a preliminary approach to the development of techniques for the automatic classification of Portuguese legal documents of the Supreme Courts and the Attorney General's office is proposed. Natural language processing techniques are combined with machine learning techniques, such as support vector machines (SVM) [10]. In Klang and Quaresma [11], the authors present a system capable of understanding the context of a user's queries in order to make suggestions for further refinement of the user's queries. They propose a classifier that receives as input a legal text and suggests a set of legal terms that characterize that text. Pisetta et al. [12] focuses on text search analysis and automatic classification of legal texts to facilitate their retrieval using linguistic tools (terminology extraction) and to determine the concepts present in the processed corpus. In the paper Quaresma and Goncalves[13], the authors discuss the problem of information extraction from legal documents using linguistic information and machine learning techniques. The interesting thing about their approach, which we have also explored in our work and which is included in our proposed methodology, is that in this approach top-level legal concepts are identified and used to classify documents using SVM, while named entities are identified using semantic information from the output of a natural language parser. This information, the legal concepts, and the named entities are then used to populate an ontology that enables document enrichment. Ontology population from text is becoming increasingly important for NLP applications. The paper Witte et al. [14] describes a GATE resource called OwlExporter that allows existing NLP analysis pipelines to be easily mapped to Ontology Web Language (OWL) ontologies, populating ontologies and

enriching them with NLP or information extracted from texts. The paper Celjuska and Vargas-Vera [15] proposes Ontosophie, a system for a semi-automatic population of ontologies with instances from unstructured text. It is based on supervised learning, learning extraction rules from annotated text and then applying these rules to new articles for the ontology population. The work reported in [16] proposes an automatic ontology population approach that uses an ontology to automatically generate rules to extract instances from text and classify them into ontology classes. These rules can be generated from the ontologies of any domain, making the proposed process domain independent. Ayadi et al. [17] presents an interesting Deep Learning-based NLP ontology population system to populate biomolecular network ontology. Bast et al. [18], Schutz and Buitelaar[19] also proposes interesting applications to semantic research and relation extraction from the text in ontology extension.

Groothuis and Svensson [20] discusses how expert systems can be used in administrative organizations to ensure legal quality. The authors of this study emphasize the value of utilizing automated reasoning tools to enhance decision-making performance, even if expert systems will never guarantee legally right conclusions because their scope and depth will always be constrained.

Finally, [21–23] discuss the application of Information and Communication Technologies (ICT) in legal decision-making by government agencies and the general applicability of legal expert systems in service delivery.

Considering the several approaches that are currently available in the literature, and the specific necessities to analyze the document text contained within juridical dossiers, we have proposed a methodology and an implementing toolchain for the recognition of entities of interest, and the identification of their relations, that exploit the existing aforementioned results, that have been tailored to the specific juridical domain. With our work, we intend to provide several functionalities, such as document organization, research, and annotation, obtained with the support of Business Process Model Notation (BPMN)-based representations. The methodology is presented in Sect. 3.

3 The proposed methodology

This section describes the methodology that we have developed for the realization of a system that can support the operations of verification and checks on documents related to the guidelines regulated by law in order to carry out proceedings. The presentation of the methodology is divided into two main parts: first, we describe the main components that make up the toolchain for semantic annotation and analysis of both BPMN workflows and documents; then, we detail the workflow of the methodology that uses these components.

3.1 A toolchain for semantic annotation and analysis of BPMNs and documents

The framework that is going to be implemented will mainly consist of four components, as shown in the unified modeling language (UML) Component diagram reported in Fig. 1. Such components are as follows:

- A **BPMN Annotator Tool** will provide a set of functions that allow uploading BPMN files in the standard BPMN 2.0 format, uploading ontologies in the OWL 2.0 format and annotating the BPMNs with concepts from the ontologies. The tool will also provide the possibility to download the annotated files so that they can be used by the other tools in the chain as a basis for inferences. This paper does not describe the BPMN annotator tool

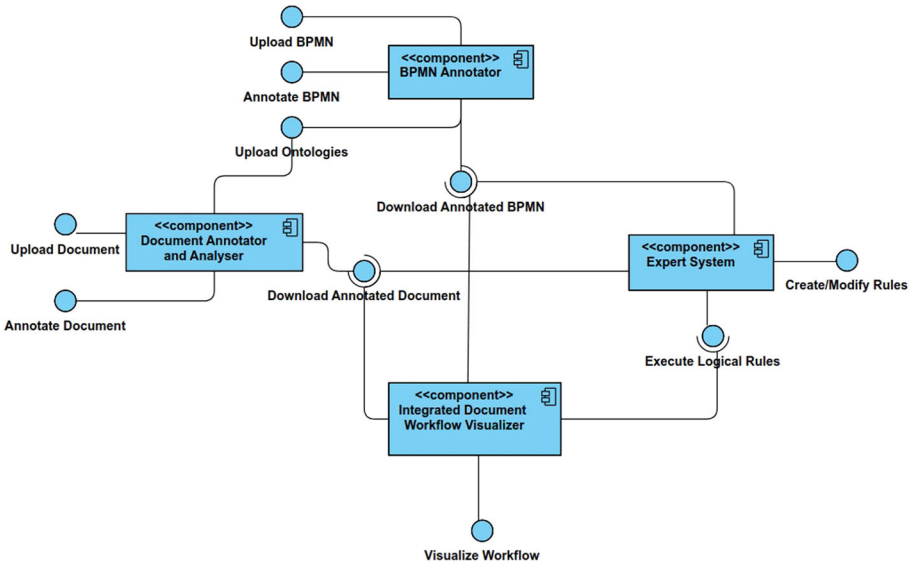


Fig. 1 Component diagram of the proposed tool chain

in detail, as it is the subject of other published work, such as [24–26]. However, Sect. 5.7 contains some basic information about such a tool.

- A **Document Annotator and Analyzer** allows users to upload documents and ontologies to be used for annotation. This particular tool allows not only the annotation of documents with specific ontologies but also their download and can be linked to the other tools in the chain to provide more complex functionalities. The upload interfaces are similar to those of the BPMN annotator tool. For more information on this specific part of the toolchain, see Sect. 5.4.
- An **Expert System** applies logical rules to the annotated BPMNs and documents, either to derive new knowledge to be stored as part of the knowledge base, or to validate them and verify specific standards applied in the domain. Sect. 5.6 provides more details on this specific tool.
- A **Integrated Document Workflow Visualizer** provides a clear view of document and BPMN annotations and allows browsing of dossiers and the visualization of links between different documents and the steps of the BPMN workflow. This tool is described in more detail in Sect. 6.

3.2 The methodology workflow

In this section, we describe the methodology we have developed for the realization of a system that supports the review and checks on documents related to the guidelines regulated by law in order to carry out proceedings. To develop this methodology, whose workflow is shown in Fig. 2, we interviewed experts in the field and the co-authors of the article to elicit their experiences and problems. Based on this, we have analyzed and developed the best solution to make the workflow more linear and efficient and to propose a solution that helps the Judge in his work.

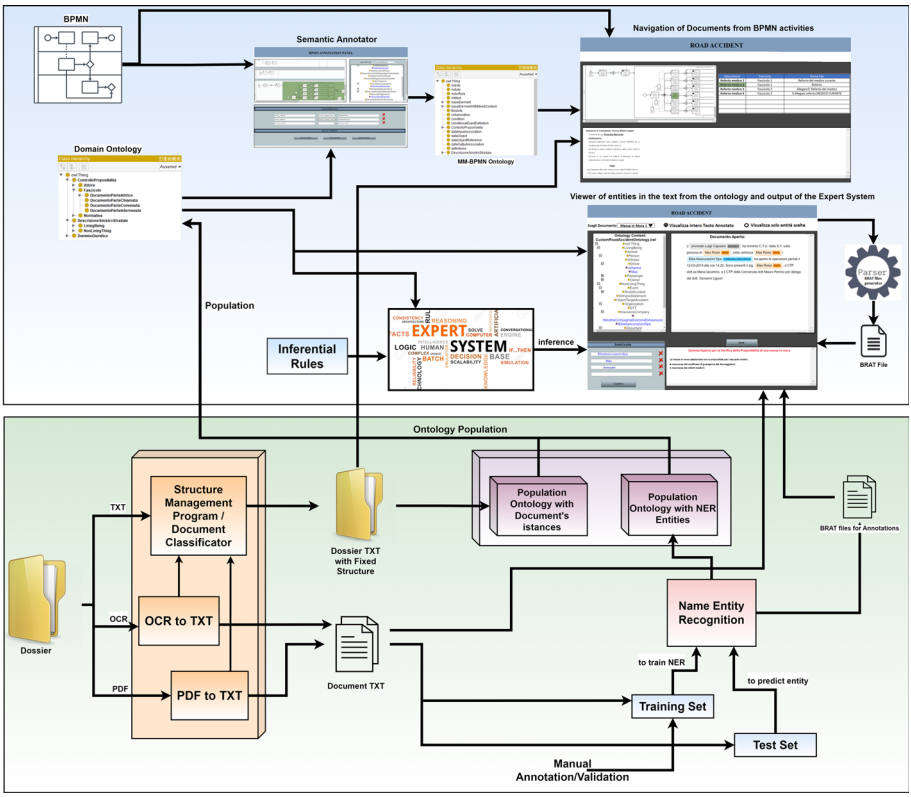


Fig. 2 Proposed workflow for the implementation of the methodology

The following are the steps into which the proposed methodology can be divided, as graphically depicted in the workflow figure:

1. Process elicitation in BPMN;
2. Ontology implementation;
3. Definition of document classification dossier structure;
4. Named Entity Recognition applied to a specific kind of document under analysis;
5. Ontology population with the outputs of the application of NLP on documents to classify them and recognize entities into them;
6. Definition of logical rules for the expert system;
7. Realization of the entity display module in documents;
8. Realization of display module for visualization of the output expert system;
9. Annotation of the BPMN to associate each BPMN Task to a document class;
10. Implementation of BPMN document navigation module.

We investigated whether it is possible to automate the document control process as much as possible to save the Judge or certain staff the time of manual and tedious operations and to ensure that automation avoids human errors that could invalidate the controls. First, with the help of experts, we elicited the process on which we wanted to focus our analysis. To do this, it was necessary to represent the process in question using a standard notation. Therefore, it was decided to represent the entire process through a **Business Process Modeling Notation**,

which was created in constant interaction with the subject matter experts. This activity is the step "**Process elicitation in BPMN**".
Then, with the help of the domain experts, a domain ontology was created, also drawing on the ontologies already available in the literature, in order to model all the concepts of the analyzed context, focusing mainly on the classes modeling the documents involved in the different processes and activities and the different actors involved in the different processes, as well as the protagonists. This activity is the step of "**Ontology implementation**"
A remarkable problem that professionals have confronted us with concerns the large number of documents they receive. These are often documents in different formats (pec documents, scans, images, texts, PDF, reports, etc.), coming from heterogeneous sources and usually not sorted or given meaningful names to facilitate retrieval. Often, the Judge or the professional staff has to verify the actual existence of certain documents provided for by law, which the lawyer or his client has to produce, before the hearing. It is obvious that this verification is very time-consuming when documents have to be consulted that are not in order and without any identification of their content. Based on the analysis of this problem, we have studied the problem of classifying documents. Professionals have pointed out that the files/dossiers available to them lack structure, which makes consultation and subsequent review tedious. Therefore, our methodology proposes a structure for the files/dossiers in which each file is a folder named after the file identifier that appears in the Judge's console. Each folder in the file contains folders whose names match the classes that the document classifier needs to recognize (e.g. recovery certificate, medical report, etc.). For each document, the classifier must recognize to which of these categories it belongs and place the file in the correct folder, keeping the original name of the file. There must be a category "OTHER" corresponding to documents that do not belong to any of the proposed categories (e.g. identity cards, invoices, etc.). We have already mentioned the heterogeneity of the formats of the documents that make up the file. In order to carry out the classification of the documents and the identification of the entities in the texts, natural language processing, and machine learning techniques must be applied, and in order to process the elements of the dossier, all the documents are in ".txt" format. We have therefore planned programs that will do this conversion of the documents into text in ".txt" format. We have also prepared a component called **"Structure manage program**" that will take care of classifying the documents and identifying the entities in the texts and return as output a "**Structured dossier**" consisting of the ordered documents in ".txt" format with meaningful labels. This activity is the step of "**Defining the structure of the dossier for document classification**".
The documents in the Dossier appropriately converted into ".txt" format, will form the training set and the test set, for the components that will apply techniques of NLP and machine learning to carry out activities of named entity recognition. In particular, since here we are considering a specific domain, it will be necessary to recognize entities within specific texts, besides the standard ones such as names of persons. To do this, it will be necessary to construct a significant training set, and this construction was foreseen an activity of manual annotation of the specific entities that one wants to recognize within the texts of a Dossier. To perform this activity of manual annotation of entities within the texts, one of the textual annotation tools for conducting NER activities that are available on the web will be used. This activity is the step of "**Named Entity Recognition applied to a specific kind of document under analysis**". The technologies and tools used for document classification and the identification of entities in the text will be illustrated more specifically in dedicated Sects. 5.4 and 5.5.
The output information of the activities of document classification and identification of entities in the text will be respectively the names specifying the type of document and the various entities recognized within the texts, these outputs will become instances of a domain

OWL ontology and, to do this, will be a program dedicated to the Ontology population with the results of the block which will carry out the operations of NER and document classification. This activity is the step of "**Ontology Population with the outputs of the application of NLP on Documents to classify them and recognize entities into them**".
We define the realization of an expert system to support able decision-making, based on the available information concerning document classification and the identification of entities within texts, which through an Ontology population activity will have become instances of an OWL Ontology and will therefore constitute our knowledge base on which it will be possible to infer through a system of inference rules, the appropriate checks. This activity is the step of "**Definition of logical rules for expert system**".
An impacting and very intuitive graphical visualization of the labeled entities recognized within texts, as well as the output of the expert system, is proposed, using a series of graphical libraries, constructing a "**Text entity viewer module**" and an " **Output expert system viewer module**". These activities are the step of "**Implementation of the entity display module in documents realization of a display module for visualizing expert system output**".
An "**Annotation program**" component will be dedicated to the semantic annotation of the aforementioned BPMN with concepts of a domain Ontology, to make explicit information that would be hidden without the application of semantics (e.g. process task mapping - actor responsible, process task mapping—documents involved, etc). The semantic annotation of BPMN with concepts of an **OWL** ontology can be done with a web-based tool, which our research group has developed and is conveniently described in [24]. This activity is the step of "**Annotation of the BPMN to associate each BPMN Task to a document class**".
We will aim to integrate into this semantic annotation tool, what is described by the proposed methodology, in particular, once the BPMN has been annotated and the OWL ontology has been populated with the results of the NER containing information on the types of documents in the Dossier and with the entities recognized within the texts, an integrated system is proposed which can visualize through a very intuitive graphic the documents within the Dossier in which the labels recognized with the NER are also shown, It will be possible to use the Ontology to explore the text selected from the constituent documents of the Dossier since the Ontology is populated with the entities recognized in the texts, it will be possible to search within the text by consulting the concepts expressed by the classes of the OWL taxonomy, (e.g. display in the text who is the damaged party, what are the income conditions, etc). On the other hand, it will be possible to use the BPMN, the visualization of which is integrated into the tool, to carry out the navigation of the Corpus by clicking on the activities of the BPMN suitably annotated semantically. This activity is the step of "**Implementation of BPMN Document Navigation Module**". The details of the technologies and techniques proposed for the implementation of the several components of the system will be discussed in more detail in the following sections.
This proposed methodology is applied to the juridical case because the texts have a very specific contextual connotation, but the flow of operations proposed, as well as the design of the various functionalities is a general purpose, which makes this methodology easily applicable to other cases and contexts with similar needs.

4 Case study: request for compensation for road traffic damage

The Italian legislation on road traffic damage is very comprehensive and mainly aims to settle disputes in the extrajudicial phase, avoiding having to go before a Judge. A large portion of

the litigation pending before the judicial offices concerns this type of dispute which has a very serial character about the legal issues that need to be examined. The legislative decree n. 209 of 2005[1] provides, first of all, in art. 145[2] that the request to the Judge to obtain compensation for damage caused by the movement of vehicles and boats, for which insurance is required, can be proposed (***condition of proposability***) only after 60 days have spent for damage to property or 90 in case of personal injury, from the date on which the injured party claimed compensation from the insurance company, by registered letter with acknowledgment of receipt having observed the methods and contents provided for in article 148.[3] Art. 148 establishes that the request must contain an indication of the personal ID of those entitled to compensation and a description of the circumstances in which the accident occurred and be accompanied, to ascertain and assess the damage by the company, data relating to the age, the work activity of the injured party, his income, the extent of the injuries suffered, a medical certificate proving the healing with or without permanent after-effects, as well as the declaration according to article 142, paragraph 2, certifying that he is not entitled to any benefits from institutions that manage compulsory social insurance or, in the event of death, from the victim's family status. The Judge is consequently required to verify the existence of these requirements and thus the existence of the *"spatium deliberandi"* of 60 or 90 days before the proposition of the judgment required by law in favor of the insurance to allow it to decide whether it intends to acknowledge the damage or ask for more information. If these requirements are not satisfied, the Judge issues a judgment of "non-proposability", which ends the trial.

5 Semantic representation

A uniform semantic representation of the domain of interest is the focus of much of this work. It was decided to use OWL to create an ontology to obtain such a representation. We were unable to locate an existing ontology that would model our particular domain and incorporate all of the concepts that were required for our analysis because the domain of interest is very large and at the same time specific to a very complex field, the legal one, applied to the specific case of proceedings for damages related to road accidents. As a result, ad hoc ontologies tailored to our situation had to be developed in conjunction with legal experts. We created two ontologies in OWL through meetings and interviews with domain expert Judge who assisted us in the implementation, these are listed as follows:

- *Juridical ontology* is an ontology that models all the concepts related to the actors involved in the juridical domain (judges, magistrates, clerks, parties, lawyers, etc.), as well as the objects and subjects of a process. A preliminary draft of this Ontology has already been presented in work [27].
- *Proposability ontology* is an ontology specific to the case of verifying the proposability of a claim for damages. It models the concepts relating to this domain in terms of the subjects involved in such proceedings (lawyer, insurance company, the injured party, etc.), data and documents (claim for damages, medical report, certificate of recovery, etc.) exchanged in such proceedings.

Since the case under analysis analyzes the procedure of claiming damages when there are road accidents, it was necessary to have a representation in OWL also of all the concepts

[1] https://www.normattiva.it/uri-res/N2Ls?urn:nir:stato:decreto.legislativo:2005-09-07;209.

[2] https://www.brocardi.it/codice-delle-assicurazioni-private/titolo-x/capo-iii/art142.html.

[3] https://www.brocardi.it/codice-delle-assicurazioni-private/titolo-x/capo-iv/art148.html.

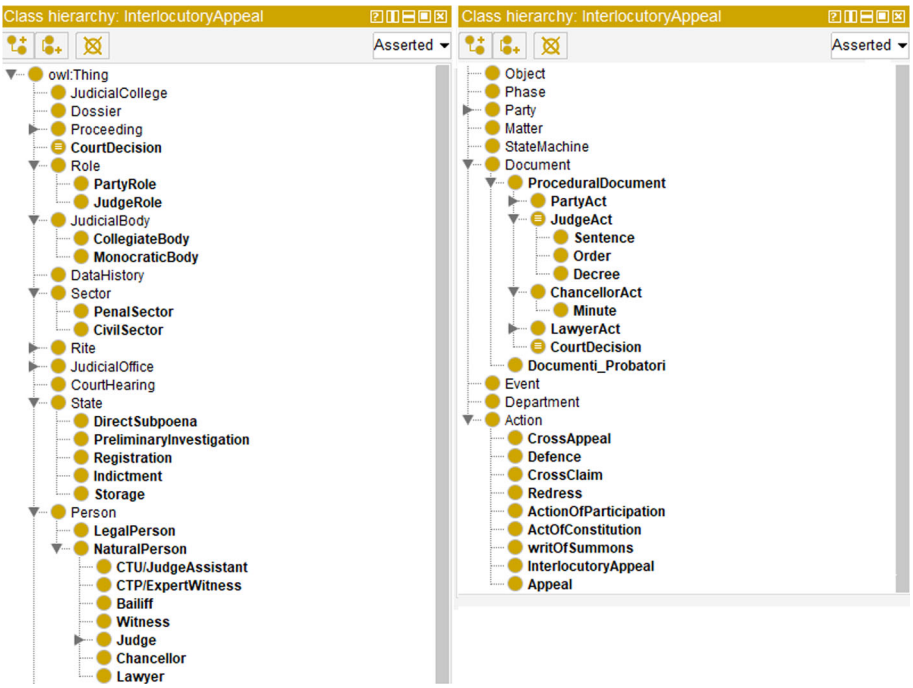


Fig. 3 Juridical ontology main classes

included in this context and, to this end, we used an Ontology found in the literature, which we expanded with other concepts we needed, this Ontology is the **Road Accident Ontology - ROA**, that semantically represent traffic accidents, their parts, location, causes, effects, etc.

These three ontologies have been combined into a single Ontology that forms the Knowledge Base (KB) on which we have been working. In dedicated Sects. 5.1, 5.2, and 5.3, more details about the above Ontologies, their main classes and properties were provided.

5.1 Juridical ontology

To model the juridical domain, in collaboration with domain experts from the Ministry of Justice we built a juridical ontology in OWL in which all actors, places, and documents related to the legal domain such as *judge*, *lawyer*, *court*, *party* are present.

For the construction of the Juridical Ontology, we have referred, for some concepts, to the ontology described in work [28], which includes the basic normative components of legal knowledge: deontic modalities, obligative rights, permissive rights, liberty rights, liability rights, different kinds of legal powers, potestative rights (rights to produce legal results) and sources of law.

Work Ceci and Gangemi [29] also has provided interesting insights into the construction of the Juridical Ontology: in particular, it describes an OWL ontology that represents the interpretations performed by a judge while conducting a discourse toward an adjudication.

Juridical ontology main classes are shown in Fig. 3.

As shown in Fig. 3, among the concepts modelled in the ontology is the class *document*, which semantically models the documents used in Italy in the Telematic Civil Trial. There are

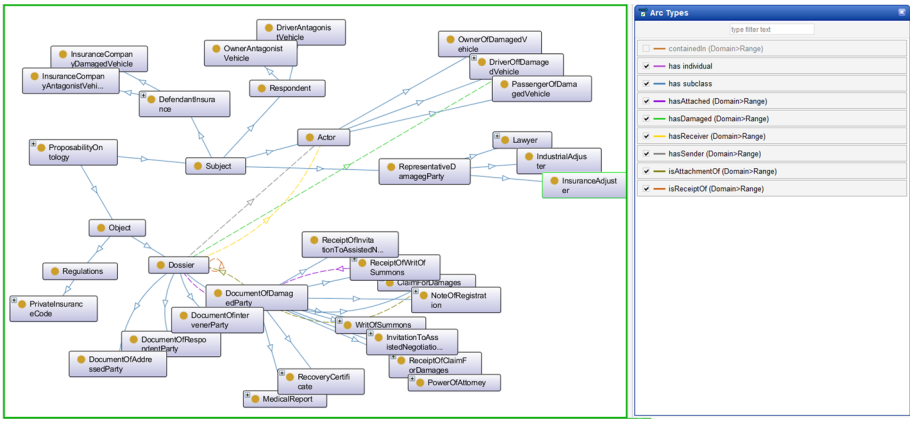


Fig. 4 Proposability ontology main classes and relations

several sub-classes of document that models the various document kinds, such as *LawyerAct*, *ChancellorAct*, and *JudgeAct*. Another very important concept contained in the ontology is *Action*, which models the various legal actions defined in the Telematic Civil Trial, such as *Appeal* e *Redress*. The *Person* class is also very important, as it describes all legal and natural persons involved in the Telematic Civil Trial. Furthermore, the ontology contains relevant concepts such as *Dossier*, *JudicialOffice*, *Rite*, *Role*, *Event* and *State*.

5.2 Proposability ontology

An analysis of the state of the art has been carried out to analyze all previous works conducted on the case study under consideration. In the literature we did not find an ontology that modeled the case of proposability, for this reason, for this work an ontology in OWL called "Proposability Ontology" has been realized, aiming at modeling the case study of the verification of the proposability of a claim for damages about traffic accidents that remain in the scope of civil procedure. This Ontology models all the concepts that were useful to implement a verification of the feasibility of a claim for damages, analyzing the regulations that govern this matter. Figure 4 shows the main classes and relations of this ontology.

Ontology consists of two main classes: *Subject* and *Object*. In the *Object* class are modeled all regulations relevant to the case study and all documents useful for the verification of the admissibility of claims for damages. Documents are divided by category; for example, the sub-class *DocumentOfDamagedParty* models documents such as medical reports, certificates of recovery, notes to register, claims for damage, and so on. The *Subject* class, on the other hand, defines all persons involved in the process. The *DefendantInsurant* class defines the insurance companies of the damaged vehicle and the antagonist vehicle. The *Actor* class defines concepts such as the owner, the driver, and also any passengers of the damaged vehicle. The *Respondent* class defines the owner, the driver, and any passengers of the opposing vehicle. Finally, the class *RepresentativeDamagedParty* defines all the figures involved in the defense of the damaged party, such as the lawyer, the industrial adjuster, or the insurance adjuster.

Several relations have been defined in the Proposability ontology, such as *hasRecipient*" and "*hasSender*", which specify that a document is addressed to a specific kind of actor or

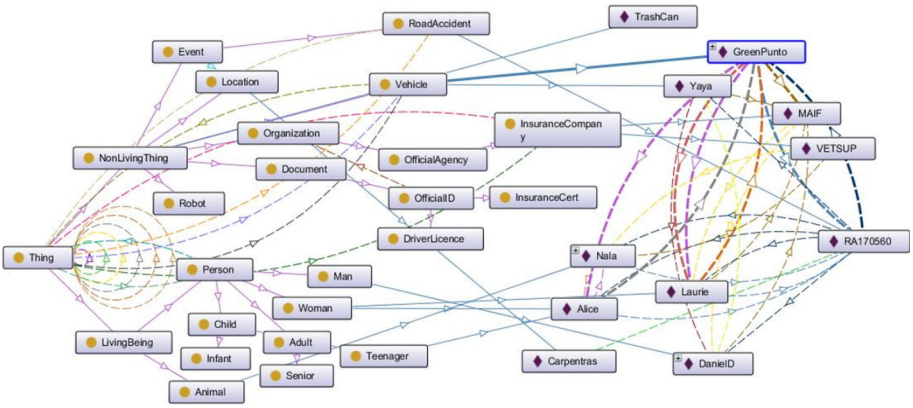


Fig. 5 ROA ontology main classes

is sent by a specific actor. Another very important property is "*isReceiptOf*, which specifies that a specific document in the Dossier is the receipt of another document.

5.3 Road accident ontology—ROA

The Road Accident Ontology, ROA,[4] was realized in OWL by Daniel Dardailler in 2012 and, in the same year, also shared at the W3C. In this ontology, we found all the concepts useful to model the domain of road accidents, which we were interested in since within a claim for damages there is information related to the road accident to which the claim refers. An image with the main classes of the ROA Ontology, using the OntoGraph plugin from Protegè, is shown in Fig. 5.

We have extended this ontology by inserting also concepts related to the ***CID***,[5] which is the model of a friendly report of the accident provided by the convention for direct indemnity. Figure 6 shows the main classes and relations of the ROA Ontology extended by us; first of all, two macro-categories can be distinguished which are the classes "**BeingLiving**" and "**NotLivingThing**". In the first class, we distinguish the two subclasses "**Animal**" and "**Person**", to the latter belong subclasses that typify more specifically the role with which a certain person is involved in an accident it includes the subclasses: "**Driver**", "**Passenger**", "**Witness**" and "**Owner**". On the other hand, the "**NotLivingThing**" contains within it the subclasses "**Document**"(which models the documents such as "*Driving License*", "*CID*", "*Insurance*", etc), "**Event**", "**Organisation**" (e.g. "*DTT*[6]", "*Insurance Company*", etc), "**Vehicle**" and "**Witness Statements**".

[4] https://www.w3.org/2012/06/rao.html.

[5] Contestazione Amichevole d'Incidente.

[6] Motorization Certificate (DTT) is a document extracted in real-time from the Civil Motorization archive. Through this type of certification, it is possible to obtain technical information relating to the vehicles (including trailers and semi-trailers) for which it is carried out.

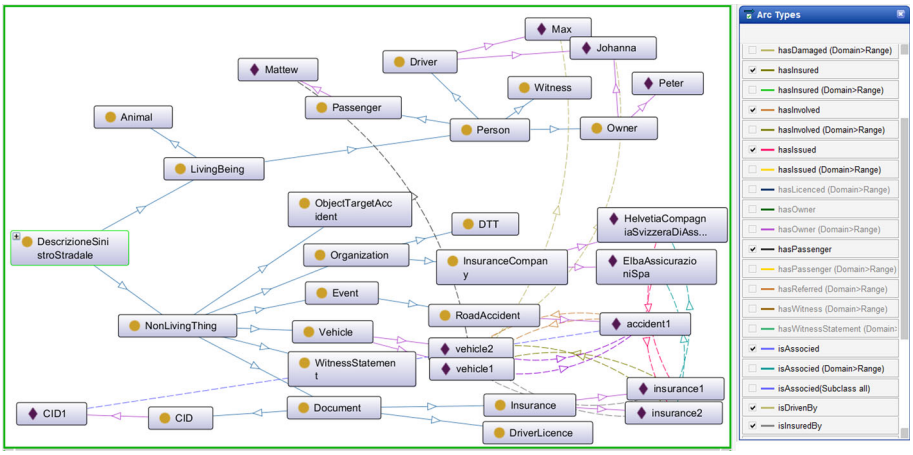


Fig. 6 ROA ontology extended classes and relations

5.4 Document classification to ontology population

The objective of document classification is to organize the dossier automatically, classifying the present documentation. For the document classification, we relied on the use of the Gensim [30] framework, and in particular, on the Doc2Vec model [31] for the training of the document classification model. Fine-tuning a Doc2Vec model with a very small dataset can be a challenging task, as there may not be enough data to effectively update the model's parameters. Here are some strategies that we use:

- *Transfer learning*: Use a pre-trained Doc2Vec model and fine-tune it on our small dataset by keeping the majority of the weights fixed and only updating a few layers. This can help to leverage the knowledge learned from a larger dataset and improve performance on a small dataset.
- *Hyperparameter tuning*: Experiment with different hyperparameters such as the number of dimensions in the vector representations, the number of training epochs, and the learning rate to find the best configuration for our specific dataset.
- **Preprocessing**: Text preprocessing is also very important, such as removing stop words, stemming, and normalizing the text.

It is important to keep in mind that it is difficult to achieve high accuracy when working with a very small dataset.

We started with preprocessing, by converting all the documents to be tagged in the text format, and then, we tagged all the documents present in the available dossiers about 10 dossiers containing about 7/10 documents belonging to different classes, for a total of 10 document classes. Once tagged, we trained the Doc2Vec model with this sets of hyperparameters: $vector\_size = [20, 50, 100]$, $window = [3, 5, 7]$, $epochs = [100, 200, 500]$. Evaluating the accuracy is a bit tricky because the model has different accuracy for different classes. For example, we take 20 different documents, in this set, there are 5 documents from class 0 and 5 documents from class 1.

- **class 0**: Precision 2/6—0.33, Recall 2/5 −0.4
- **class 1**: Precision 4/6—0.66, Recall 4/5 −0.8

Performance can also be improved with a deeper cleaning of the text and applying n-grams [32].

5.5 Ontology population with results of NLP Techniques applications

Figure 7 shows how the steps necessary to populate the Ontologies introduced in Sect. 5. All the dossiers are presented to the system in TXT format, after being preprocessed from PDF files or other text formats. The first step of the pipeline consists of the application of simple regex, aiming at recognizing elements with precise formats, such as personal ID, dates, or license plates. All the identified elements are matched with the existing ontologies, to verify if they have been previously encountered in different documents and if their semantic connections to entities and other elements are already known. Regex does not have enough power to identify all of the elements of interest: for this, NLP techniques are applied, to recognize names of people, places, and events but, most importantly, of relationships and connections existing among them. Again, all the identified entities are verified through the existing ontologies, so that existing relationships can be confirmed and can be used to support the identification of new entities.
The final step consists of the actual population of the ontology, with the enrichment of existing entities or the creation of new individuals and new relationships accordingly.

The NLP techniques present in the pipeline in Fig. 7 are the result of a further pipeline for the creation of a specific NLP model for the application domain of the case study, the legal domain. The first step is to create a dataset to use for training. For the construction of the dataset, we used two annotation tools based on a web interface, Doccano[7] and BRAT,[8] with which we annotated a small very specific dataset [33, 34]. Then, on the same dataset, we used regular expressions to extract other entities useful for training (dates, social security numbers, license plate numbers, identity document numbers, and VAT numbers). At the end of the dataset creation phase, we divided the dataset into the training set, validation set, and test set. Once the dataset was divided, we selected an NLP model for the Italian language not trained and we started the training by increasing the number of iterations. The first results obtained from the model are very promising and have allowed us to validate the accuracy of the dataset and the validity of the training pipeline. Once the pipeline was validated with the restricted dataset, we decided to use a larger and more varied dataset to have further validation and on which we will train the model for an always specific but slightly wider domain, the legal domain. We have selected a dataset of about 30,000 legal documents to be noted, but they must be noted to be used for training. Fortunately, the dataset is somewhat structured and we know the string relating to a subset of entities to be detected. In our prototype, we have identified only three specific domain entities to apply the pipeline, in such a way as to validate it on a restricted set of entities but on larger and generic datasets, always related to the specific domain. The training with the Spacy library has these parameters: iteration $= 10$, drop $= 0.35$, sdg = optimizer. The evaluation of the resulting trained model has these results: precision = 0.666, recall = 0.581, and F-measure = 0.620. These results were not good enough, so we double-checked the dataset and we found a lot of errors also on the metadata used for the automatic extraction of the training set, so the cleaning phase must be updated. We then applied a very similar pipeline for training the same model and also for identifying relationships. Through the tools mentioned above we have prepared the restricted dataset (about 50 documents), and it was used to train the NER model to extract the relationships.

[7] https://doccano.herokuapp.com/.

[8] https://brat.nlplab.org/.

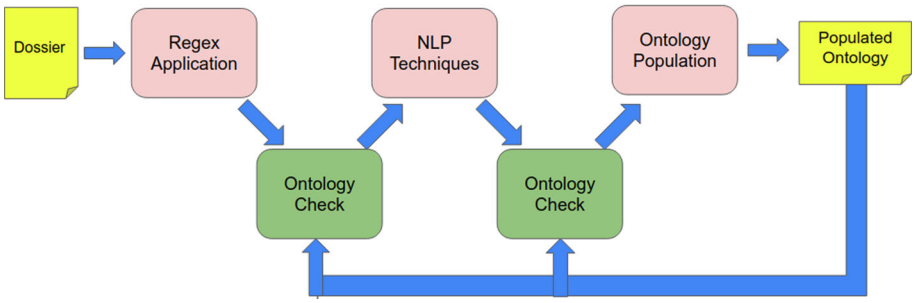


Fig. 7 The generic pipeline used to populate the Ontologies

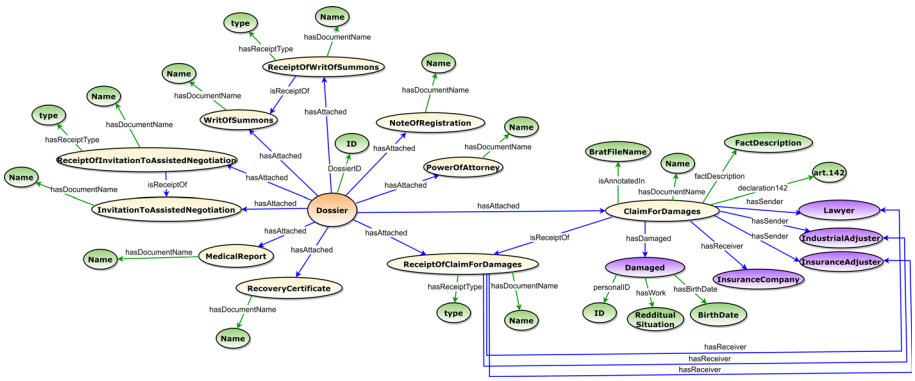


Fig. 8 Semantic network for ontology population

These first training were performed using *Word2Vec* [35] and a ruler inside the spacy pipeline. Still, we have already updated the pipeline to use *BERT-type transformers* [36], specifically for Italian, to improve performance further.

All entities and/or relationships among them recognized through the activity of named entity recognition described in 5.5 and the results of the activity of documental classification described in 5.4 are used to perform an activity of Ontology population and then populate the Ontology of proposability that we have described in dedicated Sect. 4. To emphasize in a better way the several concepts with which Ontology has been populated and to show graphically their reciprocal relationships, it is proposed a semantic network in Fig. 8. As can be seen from the figure, the classes are shown in yellow, the data properties are highlighted in green, and the object properties are in blue. In the text of the claim for damages, the injured party can be referred to with different names, such as Name-Surname, Surname-Name, or just Surname. The final ontology contains a single instance of it, and all the possible ways in which the injured party is reported are reported with a data property *hasAlias*. Only one name is chosen by the populating program, but all aliases are kept.

5.6 Expert system with semantic techniques

A system of seven inference rules written in the Semantic Web Rule Language (SWRL) [37] was implemented to perform a verification of the proposability conditions of a claim for damages. These rules are applied to a knowledge base, which is represented by the ontologies

Table 1 Final rule of expert system: verification of a claim for damages' proposability

7pc Natural language rule	SWRL language rule
A claim for damages is proposable $\Longleftrightarrow$	ClaimForDamages(?Mor) $\wedge$
It's a valid request of a claim for damages	CertificateHealing(?CertG) $\wedge$
It contains the injured party's taxpayer identification number	ReportMedical(?Med) $\wedge$ isValidRequest(?Mor,True) $\wedge$
It contains the age and income information of the injured party	containsPersonalIdDamaged(?Mor,True) $\wedge$ containsAgeDamaged(? Mor,True) $\wedge$
It contains a statement pursuant to section 142	containsDamagedIncome(?Mor,True) $\wedge$ declaration142(?Mor,?Dec) $\wedge$
It contains a description of the event	factDescription(?Mor,?Fact) $\wedge$
It contains a description of the event	hasAttached(?Mor,?Ref) $\wedge$
It is accompanied by at least one medical report	hasAttached(?Mor,?CertG) $\Rightarrow$ isProposable(?Mor,True)
It is accompanied by a certificate of recovery	

defined in Sect. 5, and are executed using the OWL DL reasoner **Pellet** [38], which follows a Forward Chaining inference method [39].

Table 1 shows the final rule of the expert system that checks if a claim for damages is proposable; to perform this checks, several conditions must be satisfied based on "*Article 148 of the Insurance Code: Compensation Procedure*". The left side shows the natural language rule, while the right side shows the SWRL rule.

This rule performs several checks to verify whether all conditions of proposability are satisfied. Some checks concern the presence of certain documents, such as healing certificates and medical reports, while others concern the presence of certain information in the claim for damages, such as the declaration according to Article 142 or the age, the personal ID, an accident's description and the financial situation of the damaged.

The execution of this rule involves other sub-rules, which perform more specific checks, such as the one shown in Table 2, that perform the verification of the validity of the claim for damages.

The rule shown in Table 2 asserts that a claim for damages is valid if and only if the document is addressed to an insurance company, the sender is an attorney or an insurance adjuster or an industrial adjuster, and there is a receipt of notice and both the claim document and the respective receipt refer to the same damaged. To implement an OR statement in SWRL, we have defined three different versions of the same rule, and for each of them, we have used a different class domain of the object property *hasSender*: in the first version the domain is *Lawyer*, and in the other versions the domains are *InsuranceAdjuster* and *IndustrialAdjuster*. Below are listed the other inferential rules implemented:

- **hasDamaged(?Rec,?Dam)**: verifies that a receipt of an claim for damages X is related to a Damaged Party Y. The check to perform are as follows: (i) There is a claim for damages Z whose receipt is X; (ii) the claim for damages Z is related to the Damaged Party Y; (iii) the claim for damages Z and its receipt X have the same sender W, which must be an instance of Lawyer, InsuranceAdjuster or IndustrialAdjuster.

Table 2 Rule for verifying the validity of a claim for damages

Natural language rule	SWRL language rule
A claim for damages is valid $\Longleftrightarrow$	ClaimForDamages(?Mor) $\wedge$
The document is addressed to an insurance company	InsuranceCompany(?IC) $\wedge$ Damaged(?Dam) $\wedge$
The sender is a lawyer or an insurance adjuster or an industrial adjuster	ReceiptOfClaimForDamages(?Rec) $\wedge$ Lawyer(?Law) $\wedge$
There is a receipt of a claim for damages and both the claim document and the respective receipt refer to the same damage	hasRecipient(?Mor,?IC) $\wedge$ hasSender(?Mor,?Law) $\wedge$ hasDamaged(?Mor,?Dam) $\wedge$ hasDamaged(?Rec,?Dam) $\Rightarrow$ isValidRequest(?Mor,True)
	ClaimForDamages(?Mor) $\wedge$
	InsuranceCompany(?IC) $\wedge$
	Damaged(?Dam) $\wedge$
	ReceiptOfClaimForDamages(?Rec) $\wedge$
	InsuranceAdjuster(?InsA) $\wedge$
	hasRecipient(?Mor,?IC) $\wedge$
	hasSender(?Mor,?InsA) $\wedge$
	hasDamaged(?Mor,?Dam) $\wedge$
	hasDamaged(?Rec,?Dam) $\Rightarrow$
	isValidRequest(?Mor,True)
	ClaimForDamages(?Mor) $\wedge$
	InsuranceCompany(?IC) $\wedge$
	Damaged(?Dam) $\wedge$
	ReceiptOfClaimForDamages(?Rec) $\wedge$
	IndustrialAdjuster(?IndA) $\wedge$
	hasRecipient(?Mor,?IC) $\wedge$
	hasSender(?Mor,?IndA) $\wedge$
	hasDamaged(?Mor,?Dam) $\wedge$
	hasDamaged(?Rec,?Dam) $\Rightarrow$
	isValidRequest(?Mor,True)

- **containsPersonalIdDamaged(?Mor,?PersID)**: verifies that a claim for damages reports the Personal ID of the damaged party;
- **containsAgeDamaged(?Mor,?Age)**: verifies that a claim for damages reports the age of the damaged party;
- **declaration142(?Mor,?Decl)**: verifies that a claim for damages reports the statement pursuant to section 142;
- **factDescription(?Mor,?Fact)**: verifies that a claim for damages reports the description of the accident;
- **containsDamagedIncome(?Mor,?Inc)**: verifies that a claim for damages reports the income situation of the damaged party;

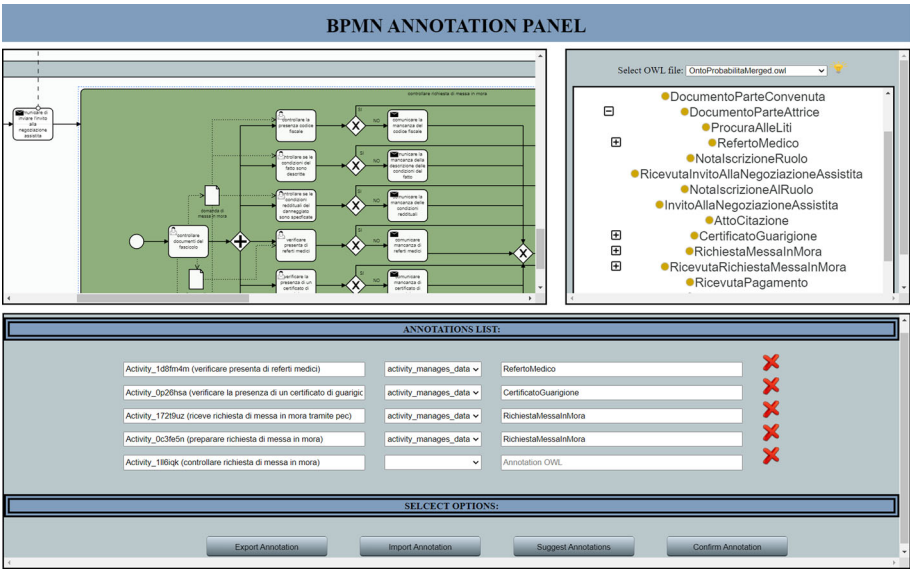


Fig. 9 BPMN semantic annotator interface panel

5.7 Semantic annotation of BPMN

To proceed with the semantic annotation of the BPMN representing the process of verifying the feasibility of a request for compensation for damages, it was necessary to create from scratch a Business Process, using the BPMN notation, which would model this case, since no existing one suited our case. For the construction of such a BPMN, we were supported by domain experts from the Ministry of Justice who described to us the internal dynamics of their offices and operations. Figure 10 shows the BPMN we have created, which has already been partly presented and described in work [4]. It helps to better understand the actors involved and the various phases of the process of verifying the proposability of a request of a claim for damages described in Sect. 4. The works [24, 26, 40, 41] provide an ad hoc methodology for semantic annotation of BPMN using Ontologies and an inferential rule-based approach and an annotation tool implementing this methodology, while an extension of this methodology that integrates security checks is presented in work [25]. Figure 9 shows the graphical interface of this annotation tool described.

From the GUI shown in Fig. 9, it is possible to visualize the BPMN on the left panel, the domain Ontology chosen to annotate the BPMN on the right panel, and all annotations inserted are shown on the bottom panel. Using the tool, it was possible to annotate each activity in the BPMN with the ontology classes that represent the kind of documents involved in the activity. The output of this semantic annotation is the "BPMN-MM Ontology", an ontology that links all structural elements of the BPMN with the domain concept using the defined annotations. Using this knowledge base, it is possible to develop a special **BPMN Document Navigation module**, which offers the possibility of navigating the various BPMN activities to display all the documents involved during them. The realized module is presented in Sect. 6.

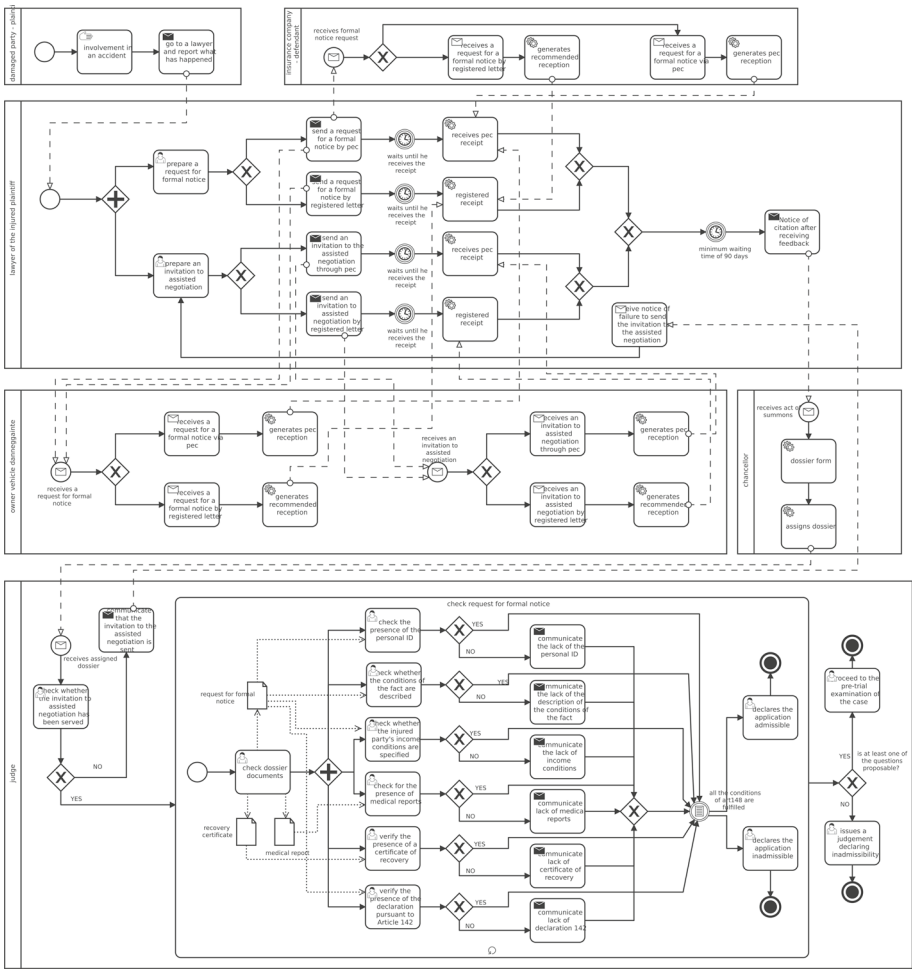


Fig. 10 Proposed BPMN for proposability

6 A prototype tool to proposability verification

To test the methodology and visualize the results with a user-friendly interface, a prototype tool was realized. Figure 11 shows the component diagram that has been created to explain the various components that will compose the prototype system. This figure also highlights the main technologies that have been used to implement the various modules.

This prototype tool is composed of three main layers, which are illustrated as follows:

- *Natural language system*: this layer is composed of a document classification module, a NER module, and an ontology population module. These modules are implemented in **Python** language, and use different technologies, such as the **Spacy** library to perform the Named Entities Activity, the document annotation tools **Brat** and **Doccano** to annotate the document to create the data set for training of NER models, the **PyTesseract** module to implement the parser PDF to TXT and OCR to TXT, the **Gensim** framework to

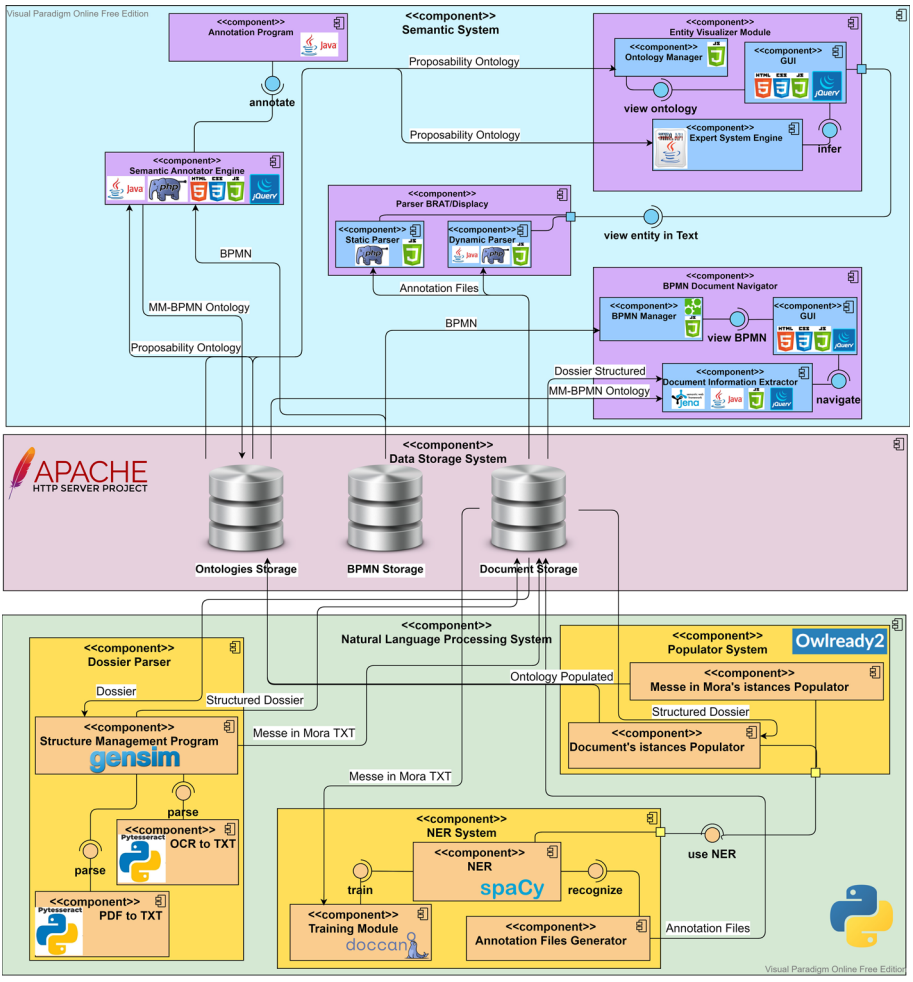


Fig. 11 Component diagram of prototype tool

implement the document classification, and the **OWLready2** module to perform the Ontology population.

- *Data storage system*: this level is responsible for maintaining the document data, the Ontologies that make up the knowledge base, and the BPMNs involved. The task of maintaining the data was entrusted to an **Apache Web Server**.[9]
- *Semantic system*: this layer is composed of different modules, such as the BPMN Semantic Annotation engine, that is responsible for BPMN semantic annotation used to create the BPMN Document Navigator module; this module is implemented using different technologies, such as **Java**, **PHP**, **Javascript**, **CSS**, **HTML**, **JQuery**, **Java RestFul Api**, and different such as **Camunda** and **OWL API**. Another very important module is the Brat/Displacy Parser, which is implemented in PHP and Javascript, and produces HTML files for the visualization of the tagged document. Another import module is

[9] https://httpd.apache.org/.

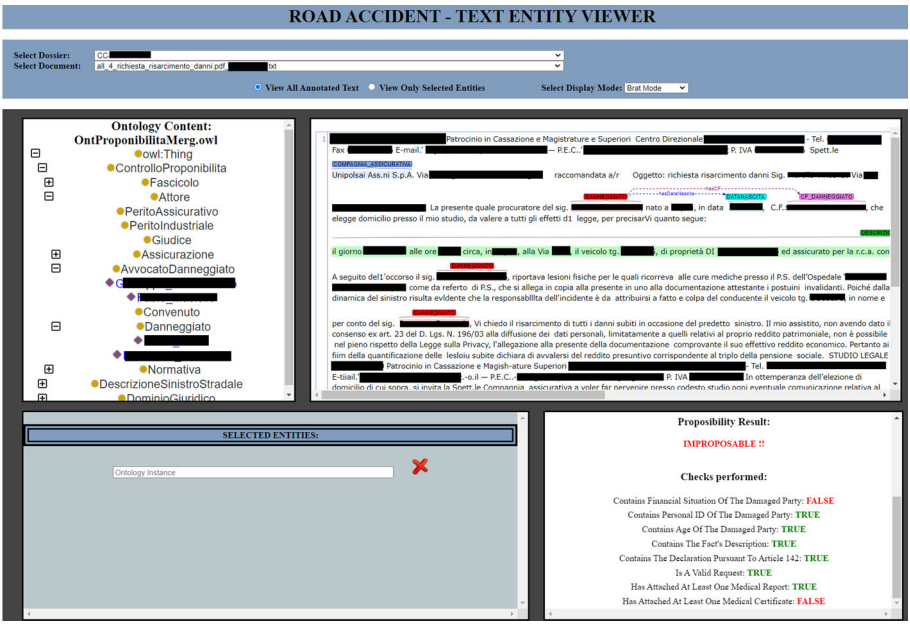


Fig. 12 Entity visualiser module: example of tagged document and output of expert system

the entity visualizer module, which is responsible for the visualization of ontologies, the tagged document, and the output of the expert system; this module is implemented using Java, JQuery, and Javascript. The last module is the BPMN document navigator, which allows one to explore and visualize the documents involved in each activity of the process, where the process is represented using the BPMN notation; this module is implemented using Java, Javascript, JQuery, the **BPMN.io API for Javascript**[10] to graphically visualize the BPMN, and **Apache Jena API**[11] to execute a query on the annotated BPMN.

Figure 12 shows the graphical interface of the entity visualizer module, from which it is possible to display the results inferred by the expert system and the NER.

For privacy reasons, references to real things, people, places, and facts have been blurred. As shown by Fig. 12, the judge can select a dossier, from which he can select a specific claim for damage, and choose appropriate display modes, and the system displays the tagged document with the entities and relations recognized by the NER, and the output of the expert system accompanied by a query explanation system that not only provides the judge with the output of the check (proposable or non-proposable) but also reports to him the output of each proposability conditions check, to provide a clear justification of the result. The tool offers two visualization modes: 'Brat mode' and 'Displacy mode', which reproduce the visualization style of the "*brat rapid annotation tool*"[12] and "*Displacy Ent tool*"[13] using the appropriate parsers. In addition, the tool offers the choice of displaying all entities and

[10] https://bpmn.io/toolkit/bpmn-js/.

[11] https://jena.apache.org/.

[12] https://brat.nlplab.org/.

[13] https://github.com/explosion/displacy-ent.

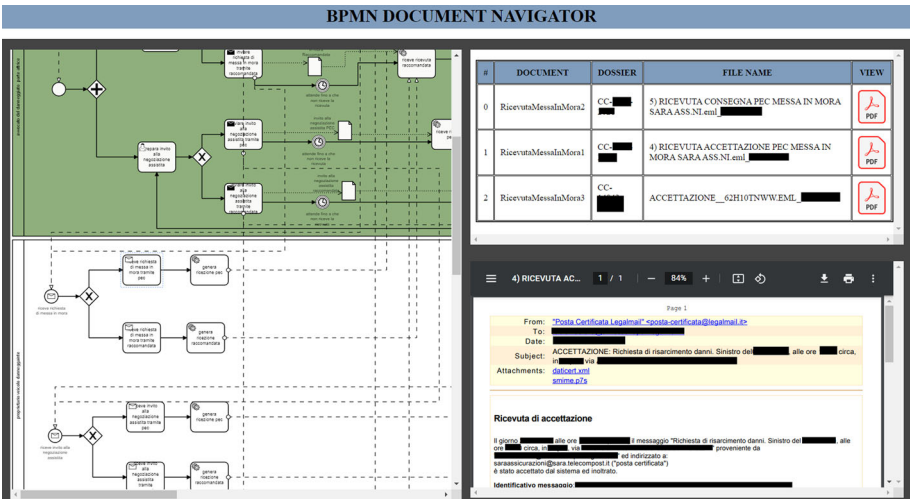


Fig. 13 Graphical interface of BPMN document navigation module

relations identified by the NER, or of displaying only a reduced set of entities by selecting them from the Ontology.

Figure 13 shows the graphical interface of the BPMN document navigation module.

As shown in Fig. 13, this module offers the possibility to "navigate" the dossiers, displaying all the documents involved in each specific phase of the process: the judge can choose one of the process activities by selecting it from the BPMN, and automatically the tool, interrogating the BPMN-MM ontology produced by the annotation program, shows the list of documents involved. In addition, the module also offers the possibility of viewing a document, downloading it, or printing it.

7 Impact evaluation

It is impossible to precisely evaluate the impact that the use of the tool and the application of the proposed methodology would have on the Trials duration. There are two main factors that influence the trial duration: the complexity of the attached dossiers, whose reading and understanding take time and is prone to human errors; the behavior of the involved parties, which is extremely variable. The presented tool acts on the Dossier analysis part, making it easier and faster for Judges to read and understand the documentation. Generally speaking, the time expended by judges and chancellors on trials strongly varies, according to a series of characteristics of the trial itself, such as the number of parties involved in the trial, the number of attorneys that represent each party, the general organization of the proposed documentation, the need to evaluate the reports of external experts and so on.

However, while these variables cannot be exactly predicted, a rough estimation of the duration of the main activities performed by Judges during trials is still feasible, if we consider a specific kind of procedure such as the Compensation Request described in Sect. 4.

Indeed, according to Domain Experts, three main activities should be taken in consideration:

1. Time to analyze each dossier and understand its content (Dossier Time—DT). This comprehends recognizing all the parties, their roles, and their representing attorneys, together with the specific object of the dossier. Domain experts report an average time of 15 min

to complete this task, but the actual range can vary from a couple of minutes to a whole hour, depending on the number of involved parties and attorneys.
2. Time expended to retrieve all the attached documentation (Retrieving Time—RT). Domain experts report an average time of 5 min, but even here it all depends on how the attorneys have presented all the documents: some attorneys are meticulous and provide very well-ordered and presented papers; others can be more disorganized and leave the burden to correctly classify all documentation on the judges' shoulders.
3. Time to analyze each document and approve/reject it (Document Approval Time—AT). Experts have estimated that a judge spends a couple of minutes to decide if a document can be admissible for the specific trial, so the overall time also depends on the number of documents (N) that have been presented.

The trial evaluation time (TET) can be expressed, in terms of the three activities that have been just described, as:

$$\mathrm{TET} = \mathrm{DT} + \mathrm{RT} + N * \mathrm{AT}$$

The proposed methodology and tool can reduce the time in all of these three aspects. In particular, activities 2 and 3 become completely automated: the tool automatically sorts the documents and presents them to the Judge, and it performs an admissibility check through the inference rules that are presented in Sect. 5.6.
Regarding activity 1, the DET has been reduced by the tool thanks to the automatic recognition of parties and related attorneys that it can perform, and to the identification of the dossiers' objects and classification. This does not mean that the time to read the dossier is reduced to zero, as the judge still has to read the motivations of the trials and the defense/offense explanation. However, the Domain Experts have estimated the overall time to be at least halved, as knowing the names and roles of parties beforehand greatly speeds up the reading and comprehension process.
In the end, the TET obtained by using the tool becomes

$$\mathrm{TET} = \mathrm{DT}/2$$

As said before, these are rough estimations, made thanks to the knowledge of the domain experts. In order to obtain exact measurements, the tool should be experimentally used by judges and chancellors in their everyday activities, which is one of the future activities that are planned.

8 Conclusion and future works

In this paper, a general methodology for the analysis of textual Documents, their classification, ontology extraction, and population, has been described, and details regarding the NLP, NER, and Regex-based techniques used to identify entities within unstructured texts have been provided. BPMN has been used to describe the phases of Trials, and its semantic annotation has been exploited to connect the documentation to the specific steps that are followed by judges and parties involved in the trials. In particular, the methodology has been applied to a specific case study, related to road accident trials and the compensation requests generally involved in them, which has been used to demonstrate the feasibility of the approach and its capability to support judges in examining the documentation accompanying each trial. The final objective is not only to reduce the duration of trials by providing Judges with a support tool for the analysis of documents and their correlation but also to implement new

functionalities such as the identification of parties involved in multiple trials or the recognition of specific relations among different parties that could help in developing new statistics and to eventually detect frauds.
In future works, the expert system that is being developed on top of the semantic representation of documents and entities identified in them will be completely implemented and used to semi-automatize the decision of Judges.
In particular, a prototype of the tool will be experimentally evaluated by selected judges and chancellors, in order to better evaluate the impact it would have on their work. As of now, only a rough estimation of such an impact is possible, based on the judgment of Domain Experts, and a more precise and punctual evaluation is needed.

Acknowledgements The work described in this paper has been funded by the Applied Research Projects "Big data Giustizia e Datawarehouse" and " Metodologie e Tecniche Innovative per la Gestione del Ciclo di Vita del Software" promoted by the Italian Ministry of Justice and realized by Consorzio Interuniversitario Nazionale per l'Informatica (CINI), and by the Project VALERE "SSCeGov-Semantic, Secure and Law Compliant e- Government Processes", granted by the University of Campania "Luigi Vanvitelli" under the VALERE:2019 program.

Author Contributions All authors contributed equally to this work and have worked on the design of the paper. BDM supervised the writing of the paper and the design of the proposed methodology and solution. LCC and MG developed the methodology and described it. SD described the application of the NLP and NER techniques within the methodology. AE focused on the design and development of the semantic aspects applied within the methodology and contributed to its description. RA and PL provided and described the Case Study, and followed all the activities regarding its management.

Funding Open access funding provided by Università degli Studi della Campania Luigi Vanvitelli within the CRUI-CARE Agreement.

Data availability The datasets generated during and/or analyzed during the current study are not publicly available but can be provided in the pseudonymized form by the Italian Ministry of Justice on reasonable request.

Declarations

conflict of interest The authors declare no conflict of interest.

References

1. Di Martino B, Colucci Cante L, Esposito A, Lupi P, Orlando M (2021) Supporting the optimization of temporal key performance indicators of italian courts of justice with OLAP techniques. In: Conference on complex, intelligent, and software intensive systems. Springer, pp 646–656
2. Di Martino B, Cante LC, Esposito A, Lupi P, Orlando M (2021) Temporal outlier analysis of online civil trial cases based on graph and process mining techniques. Int J Big Data Intell 8(1):31–46
3. Di Martino B, Esposito A, Colucci Cante L (2021) Multi agents simulation of justice trials to support control management and reduction of civil trials duration. J Ambient Intell Hum Comput, 1–13
4. Di Martino B, Colucci Cante L, D'Angelo S, Esposito A, Graziano M, Ammendolia R, Lupi P (2022) Semantic based knowledge management in e-government document workflows: a case study for judi-

ciary domain in road accident trials. In: Computational intelligence in security for information systems conference. Springer, pp 435–445
5. Di Martino B, Marulli F, Graziano M, Lupi P (2021) Prettytags: an open-source tool for easy and customizable textual multilevel semantic annotations. In: Conference on complex, intelligent, and software intensive systems. Springer, pp 636–645
6. Di Martino B, Colucci Cante L, Graziano M, Enrich Sard R (2020) Tweets analysis with big data technology and machine learning to evaluate smart and sustainable urban mobility actions in Barcelona. In: Conference on complex, intelligent, and software intensive systems. Springer, pp 510–519
7. Bisong E (2019) Logistic Regression. Apress, Berkeley, CA, pp 243–250. https://doi.org/10.1007/978-1-4842-4470-8_20
8. Assefi M, Behravesh E, Liu G, Tafti AP (2017) Big data machine learning using apache spark mllib. In: 2017 IEEE international conference on big data (Big Data), pp 3492–3498 . https://doi.org/10.1109/BigData.2017.8258338
9. Gonçalves T, Quaresma P (2003) A preliminary approach to the multilabel classification problem of portuguese juridical documents. In: Portuguese conference on artificial intelligence. Springer, pp 435–444
10. Gonçalves T, Quaresma P (2004) The impact of NLP techniques in the multilabel text classification problem, pp 424–428
11. Klang M, Quaresma P (2000) Automatic classification and intelligent clustering for wwweb information retrieval systems
12. Pisetta V, Hacid H, Zighed DA (2005) Automatic juridical texts classification and relevance feedback. Mining Complex Data, 81
13. Quaresma P, Gonçalves T (2010) Using linguistic information and machine learning techniques to identify entities from juridical documents. Semantic Processing of Legal Texts, pp 44–59
14. Witte R, Khamis N, Rilling J (2010) Flexible ontology population from text: the owlexporter. In: Proceedings of the seventh international conference on language resources and evaluation (LREC'10)
15. Celjuska D, Vargas-Vera M (2004) Ontosophie: A semi-automatic system for ontology population from text. In: International conference on natural language processing (ICON), p 60
16. Faria C, Serra I, Girardi R (2014) A domain-independent process for automatic ontology population from text. Sci Comput Program 95:26–43
17. Ayadi A, Samet A, de Beuvron FdB, Zanni-Merk C (2019) Ontology population with deep learning-based NLP: a case study on the biomolecular network ontology. Procedia Comput Sci 159:572–581
18. Bast H, Bäurle F, Buchhold B, Haussmann E (2012) Broccoli: semantic full-text search at your fingertips. arXiv:1207.2615
19. Schutz A, Buitelaar P (2005) Relext: A tool for relation extraction from text in ontology extension. In: International semantic web conference. Springer, pp 593–606
20. Groothuis M, Svensson J (2000) Expert system support and juridical quality. In: Breuker J, Leenes RE, Winkels R (eds) Legal knowledge and information systems legal knowledge and information systems. IOS Press, Netherlands, pp 1–10
21. Svensson JS (2002) The use of legal expert systems in administrative decision making. In: Electronic government: design, applications and management, pp 151–169
22. Groothuis M (2007) Applying icts in juridicial decision making by government agencies. In: Encyclopedia of digital Government, pp 87–96
23. Pethe VP, Rippey CP, Kale LV (1989) A specialized expert system for judicial decision support. In: Proceedings of the 2nd international conference on artificial intelligence and law, pp 190–194
24. Di Martino B, Cascone D, Colucci Cante L, Esposito A (2021) Semantic representation and rule based patterns discovery and verification in eprocurement business processes for egovernment. In: Conference on complex, intelligent, and software intensive systems. Springer, pp 667–676
25. Rak M, Granata D, Di Martino B, Colucci Cante L (2022) A semantic methodology for security controls verification in public administration business processes. In: Computational intelligence in security for information systems conference. Springer, pp 456–466
26. Di Martino B, Graziano M, Colucci Cante L, Esposito A, Epifania M (2022) Application of business process semantic annotation techniques to perform pattern recognition activities applied to the generalized civic access. In: Computational intelligence in security for information systems conference. Springer, pp 404–413
27. Di Martino B, Cante LC, D'Angelo S, Esposito A, Graziano M, Marulli F, Lupi P, Cataldi A (2022) A big data pipeline and machine learning for uniform semantic representation of data and documents from it systems of the Italian ministry of justice. Int J Grid High Perform Comput 14(1):1–31
28. van Engers T (2006) An owl ontology of fundamental legal concepts. In: Legal knowledge and information systems: JURIX 2006: the nineteenth annual conference, 152, 101. Ios PressInc

29. Ceci M, Gangemi A (2016) An owl ontology library representing judicial interpretations. Semant Web 7(3):229–253
30. Řehůřek R, Sojka P (2011) Gensim-statistical semantics in python. Retrieved from genism. org
31. Lau JH, Baldwin T (2016) An empirical evaluation of doc2vec with practical insights into document embedding generation. In: Proceedings of the 1st Workshop on Representation Learning for NLP. Association for Computational Linguistics, Berlin, pp 78–86. https://doi.org/10.18653/v1/W16-1609
32. Robertson AM, Willett P (1998) Applications of n-grams in textual information systems. J Doc 54(1):48–67
33. Stenetorp P, Pyysalo S, Topić G, Ohta T, Ananiadou S, Tsujii J (2012) Brat: a web-based tool for NLP-assisted text annotation. In: Proceedings of the Demonstrations at the 13th conference of the european chapter of the association for computational linguistics, pp 102–107
34. Yimam SM, Gurevych I, Eckart de Castilho R, Biemann C (2013) WebAnno: A flexible, web-based and visually supported system for distributed annotations. In: Proceedings of the 51st annual meeting of the association for computational linguistics: system demonstrations. Association for Computational Linguistics, Sofia, Bulgariapp, pp 1–6 . https://aclanthology.org/P13-4001
35. Ma L, Zhang Y (2015) Using word2vec to process big text data. In: 2015 IEEE international conference on big data (Big Data), pp 2895–2897. https://doi.org/10.1109/BigData.2015.7364114
36. Acheampong FA, Nunoo-Mensah H, Chen W (2021) Transformer models for text-based emotion detection: a review of bert-based approaches. Art Intell Rev 54:5789–5829
37. O'Connor MJ, Knublauch H, Tu SW, Musen MA (2005) Writing rules for the semantic web using swrl and jess
38. Sirin E, Parsia B, Grau BC, Kalyanpur A, Katz Y (2007) Pellet: a practical owl-dl reasoner. J Web Semant 5(2):51–53. https://doi.org/10.1016/j.websem.2007.03.004
39. Hayadi BH, Bastian A, Rukun K, Jalius N, Lizar Y, Guci A (2018) Expert system in the application of learning models with forward chaining method. Int J Eng Technol 7(2.29):845–848
40. Di Martino B, Colucci Cante L, Esposito A, Graziano M (2023) A tool for the semantic annotation, validation and optimization of business process models. Softw Pract Exp. https://doi.org/10.1002/spe.3184
41. Di Martino B, Graziano M, Colucci Cante L, Ferretti G, De Oto V (2022) A semantic representation for public calls domain and procedure: housing policies of campania region case study. In: Computational intelligence in security for information systems conference. Springer, pp 414–424

Beniamino Di Martino is Full Professor at the University of Campania (Italy) and Adjunct Professor at Asia University (Taiwan) and University of Vienna (Austria). He is author of 14 international books and more than 300 publications in international journals and conferences; has been Coordinator of EU funded FP7-ICT Project mOSAIC, and participates to various international research projects; is Editor / Associate Editor of seven international journals and Editorial Board Member of several international journals; he is vice Chair of the Executive Board of the IEEE CS Technical Committee on Scalable Computing; is member of: IEEE WG for the IEEE P3203 Standard on Cloud Interoperability, IEEE Intercloud Testbed Initiative, IEEE Technical Committees on Scalable Computing (TCSC) and on Big Data (TCBD), Cloud Standards Customer Council, Cloud Computing Experts' Group of the European Commission.

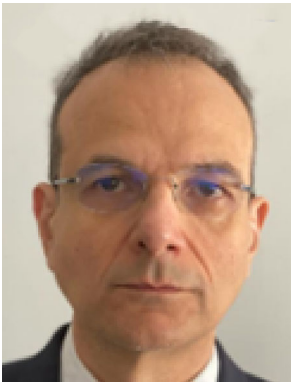
Luigi Colucci Cante is currently pursuing a PhD in Industrial Engineering and Information Technology at the University of Campania Luigi Vanvitelli. He has been Research Fellow at the Department of Engineering involved to the VALERE project "SSCeGov - Semantic, Secure, and Law-Compliant e-Government Processes." He has been involved in the "Big Data Giustizia e Datawarehouse" project sponsored by the Italian Ministry of Justice and implemented by Consorzio Interuniversitario Nazionale per l'Informatica (CINI), and in the "Horizon 2020 project GreenCharge." He graduated in Computer Engineering in 2020. Co-chiar of the 14-th International Workshop on Semantic Web/Cloud and Intelligent Systems Management (SWISM-2024). Track chair of UbiSec 2024 (The 4th International Conference on Ubiquitous Security). Co-author of several scientific articles. His main research interests are Process mining, Artificial Intelligence, Semantics, BPMN optimization and patterns Discovery, Software and Knowledge Engineering and Big Data and Cloud Computing.

ANTONIO ESPOSITO is currently Research Fellow at the Department of Engineering of the University of Campania "Luigi Vanvitelli". His PhD thesis focused on the recognition and application of Design and Cloud Patterns to Software development in Cloud Environment, with the support of Semantic Technologies. He has been involved in the EU funded FP7-ICT Project mOSAIC and in the Horizon 2020 Project Toreador, and he is currently involved in the Horizon 2020 project GreenCharge and in the Applied Research Project "Big data Giustizia e Datawarehouse" promoted by the Italian Ministry of Justice as part of the Consorzio Interuniversitario Nazionale per l'Informatica (CINI). His main interests are Software Engineering, Cloud Computing, Design and Cloud patterns, and Semantic based information retrieval.

MARIANGELA GRAZIANO is currently a PhD student in Industrial Engineering and Information Technology at the University of Campania Luigi Vanvitelli. She has been Research Fellow at the Department of Engineering on the Project VALERE "SSCeGov - Semantic, Secure and Law Compliant e-Government Processes". She has participated in projects "Legal Bim and Digital Transition," "Big Data Giustizia e Datawarehouse" by the Italian Ministry of Justice and realized by Consorzio Interuniversitario Nazionale per l'Informatica (CINI), and "Horizon 2020 project GreenCharge". In 2020, she graduated with a Master's Degree in Computer Engineering. Co-chiar of The 14-th International Workshop on Semantic Web/Cloud and Intelligent Systems Management (SWISM-2024). Trackchair of UbiSec 2024 (The 4th International Conference on Ubiquitous Security). Member of the program committee of 2021 IEEE CSR Workshop on Resilient Artificial Intelligence (RAI). Co-author of several scientific articles. Her research interests are Natural Language Processing, Deep Learning, Machine Learning, Semantics, BPMN optimization and patterns Discovery, Software and Knowledge Engineering and Big Data.

PIETRO LUPI graduated in Law at the University of Naples, Federico II, in 1987 and became judge in 1992. He was first appointed to the Santa Maria Capua Vetere Tribunal (Caserta) and then to Naples Tribunal in 1996. From February 2013 to September 2015, he was the director of IT for the tribunal supervising the IT transition following the introduction of the electronic civil trials in Italy. From October 2015 to July 2018, he was a member of the Technical Organizational Structure (STO) of the Italian Superior Council of Judges in Rome where he studied the organization of judicial offices and their performance in terms of judicial procedures and efficiency. From July 2018 to February 2021, he was seconded to the Italian Ministry of Justice in Rome where he coordinated and oversaw the implementation of IT and automated IT Services in civil procedures for the whole of Italy. In February 2021 he returned to the Court of Naples. Since September 2022 he has been president of a civil court of the tribunal. He has been an instructor on numerous courses on the application of IT to civil trials at the Italian Higher School of the Magistracy in Florence and was the organizer of the School's first course on electronic civil trials in 2014.

SALVATORE D'ANGELO is currently Research Fellow at the Department of Engineering of the University of Campania "Luigi Vanvitelli". His PhD thesis focused on parallelizing compiler techniques for multi-Cloud and Big Data platforms. During his master's degree he started participating in research projects and discovered his passion for scientific research. He participated in research projects supported by international and national organizations, such as Horizon 2020 - Toreador project, RASTA project, and in the Applied Research Project "Big data Giustizia e Datawarehouse" promoted by the Italian Ministry of Justice as part of the Consorzio Interuniversitario Nazionale per l'Informatica (CINI). Track chair at The 11-th International Workshop on Cloud Computing Project and Initiatives (CCPI 2024) and at The 4th International Conference on Ubiquitous Security(UbiSec 2024). His interests include research activities dealing with Big Data, Machine Learning, Cloud Computing, Compilers, High-Performance Computing, and Quantum Computing.



ROSARIO AMMENDOLIA in the judiciary since 2004, has mainly dealt with civil law, bankruptcy and forced executions at the Court of Savona and at the Court of Genoa, where he also held the position of reference magistrate for innovation (Magrif). Since 2021 out of position at the Ministry of Justice, he has been coordinating magistrate of the civil area of the Directorate General of Automated Information Systems and since 2023 he has been coordinator, at the office of the Head of Department for the Department for the Digital Transition of Justice, Statistical Analysis and Cohesion Policies, of the working group for international projects. Since 2022, he has been participating in the work of the e-Justice Group at the EU Council, where he negotiated for Italy the "digitalisation package" of cross-border cooperation in civil and criminal matters. He is a member of the e-Codex and Jits Collaboration Platform advisory groups at the "European Agency for the Operational Management of Large-Scale IT Systems in the Area of Freedom, Security and Justice" (eu-LISA).

Authors and Affiliations

Beniamino Di Martino[1,2,3,6] · Luigi Colucci Cante[1] · Mariangela Graziano[1] · Salvatore D'Angelo[1,6] · Antonio Esposito[1,6] · Pietro Lupi[4] · Rosario Ammendolia[5]

Luigi Colucci Cante
luigi.coluccicante@unicampania.it

Mariangela Graziano
mariangela.graziano@unicampania.it

Salvatore D'Angelo
salvatore.dangelo@unicampania.it

Antonio Esposito
antonio.esposito@unicampania.it

Pietro Lupi
pietro.lupi@giustizia.it

Rosario Ammendolia
rosario.ammendolia@giustizia.it

1 Department of Engineering, University of Campania "L. Vanvitelli", 81031 Aversa, Italy

2 Department of Computer Science and Information Engineering, Asia University, Taichung, Taiwan

3 Department of Computer Science, University of Vienna, Vienna, Austria

4 Judge of Court of Naples, Ministry of Justice, Italy

5 Department for the digital transition of justice, statistical analysis and cohesion policies Office of the Head of Department– Coordinator WG, Via Crescenzio 17/c- 00193, Rome, Italy

6 CINI - Consorzio Interuniversitario Nazionale per1 l'Informatica Via Ariosto, 25 00185, Roma, Italy