OMTAT annotation tool: semantical enrichment for legal document search *

Sylvie Szulman
LIPN, Université Paris 13
Sorbonne Paris Cité & CNRS
France
ss@lipn.univ-paris13.fr

François Lévy
LIPN, Université Paris 13
Sorbonne Paris Cité & CNRS
France
fl@lipn.univ-paris13.fr

Eve Paul
LuapLab
France
eve.paul@luaplab.com

Abstract

This paper describes a help system for legal document searching. The proposed approach relies on creating specific annotations over a corpus of documents. A tool has been built which implements the visualization of annotations, texts and semantic resources, the creation of annotations and their collation in resources. A search engine has been implemented as well to query the set of annotated documents in order to answer user questions.

1 Introduction

This paper presents a system intended to facilitate the access to French legal documents. Given the huge amount of existing legal documents, we developped a document search system for non lawyer users (trade unionist, human resources manager, etc.) to find relevant information in the overabundance of documents. Even if there are legal databases and law data Banks for the French law, most of them target professional users, so they are hard to access for the non lawyer user. Our system should be a base to make the documentation more accessible to non lawyers. The domain of experiences is restricted to the *Code du Travail* (labour code) and collective labour agreements.

The proposed approach relies on creating specific annotations over the reference documents. “Annotation” is used here in an NLP sense (a specific mark in the text) and not in its legal sense (a comment mainly based on jurisprudential cases). The work relies on the textual annotation standoff format defined by the Brat tool ((Stenetorp et al., 2012b; Stenetorp et al., 2012a)) which has been used in particular in the BioNLP domain. The approach is supported by a tool that we have built, which allows user to annotate documents with our different kinds of annotations and to query the set of documents and annotations. The tool thus enables user to rapidly find answers to a legal question in the domain.

General ideas are illustrated with an example where our approach may help a user to extract the most relevant legal excerpts for his problem. Consider the case of a professional newsman which regularly works for a journal as a freelance, so he is paid by the piece. Suddenly the journal ceases giving him work. He wants to know if he has some right to an indemnity. The base document to query is the labour code, which is 1800 pages long. This use case is cited as an example all along the paper, and is specifically considered in section 6.

The rest of the paper is divided as follows. In section 2, the approach is positioned with respect to other approaches for the access to legal documentation. Section 3) presents the types of annotations that have been defined in order to give an account of legally significant properties of the text. The OMTAT tool which has been built to visualize, explore or add annotations is described in section 4. Section 5 is specifically devoted to the query engine. The 6th and last section contains a short description of the use case.

*This work is part of the program "Investissements d’Avenir" overseen by the French National Research Agency, ANR-10-LABX-0083

2 Accessing Documentation in the legal Domain

2.1 Information Retrieval

Information Retrieval in the legal domain consists in retrieving law articles and case law decisions related to a given subject matter. The search can be by reference, for instance the “Loi Aubry” or “loi sur les 35 heures” (35hours working week law). It can also be a search of keywords in plain text, or in some specific sites of a previously defined structure. Advanced search options allow to search by document descriptors (previously annotated with the help of semantic ressources, like a thesaurus).

Major law-editors (Dalloz[1], Editions Francis Lefebvre [2], Lamy[3] , Lexis-Nexis[4]) offer to clients a large quantity of documents and an engine customized for their documents and their descriptors. Their services are accessed mostly by professionals, due to their commercial offers. Some smaller law editors make available, in the *Code du Travail* domain, mementos and/or fact-sheets that can be bought by non-lawyers economic players (named NLEP in the following): trade-unionists, human resources managers, managers of small business, etc. NLEP can then refer back to these summaries to answer practical questions. Our present approach proposes to make sources of law available for NLEP, through a semantic and structural search in documents which have been previously semantically and structurally annotated.

2.2 Semantic Approach

Information Retrieval in the legal domain has recently interested the academic community. (Berry et al., 2012) consider how different language models of the collection succeeded or failed to be used by domain specialists, (Mimouni, 2015) studies two approaches for querying a collection, viewed as a network of documents. The first is based on Formal Concept Analysis, the second on semantic web technologies, namely an ontology to annotate the collection and Sparql to query it.

In a semantic approach as the one adopted here, semantic resources, i.e. ontologies, thesauri or terminologies in the legal domain, play an important role. Resources of this kind exist but, due to business reasons, they are mostly not public.

Among public thesauri, EuroVoc is a multilingual thesaurus produced by the European Union[5]. It describes the terminology used by the different domains of activity of this Union and is available in twenty three languages. It is used, among others, by the European Parliament, the Publications Office of the European Union, some national and regional parliaments in Europe, as well as by national administrations and private users in different states, some members of the Europeaan Union and some not members.

Jurivoc[6] is a public, trilingual thesaurus used in the Swiss Confederation. It was produced by the Swiss Federal Court and the Insurance Federal Court.

These resources cope with domains other than the labour code and cannot be used as such for annotating the French *Code du Travail*. A terminology of French employment law is in construction by one of the authors who is a legal expert. Currently the built terminology is restricted to the use case. It allows to create a set of terminological annotations.

2.3 Standard based Initiatives

XML (eXtensible Markup Language) is a metalanguage allowing to define for a particular domain a set of semantic and structural marks. Several XML standards have been defined for the encoding of legal documents, like MetaLex and AkomaNtoso.

- MetaLex has been devised by the CEN as a Workshop Agreement which standardizes the way in which sources of law and references to sources of law are to be represented in XML. It involves an Open XML Interchange Format for Legal and Legislative Resources, so as to avoid locking a client to a provider (Boer et al., 2008).

- AkomaNtoso (Palmirani et al., 2005) defines a set of simple technology-neutral electronic representations in XML format of parliamentary, legislative and judiciary documents.

[1] http://www.editions-dalloz.fr/

[2] http://www.efl.fr/

[3] http://www.wkf.fr/accueil.html

[4] http://www.lexisnexis.fr/

[5] http://eurovoc.europa.eu/drupal/?q=fr

[6] http://www.bger.ch/fr/index/juridiction/jurisdiction-inherit-template/jurisdiction-jurivoc-home.htm

Note that XML standards are not exclusive of our approach, for semantic and structural markups can be converted into annotations. An advantage of our approach is the possibility to create relational annotations, that is annotations bearing on other annotations which are difficult to represent in XML.

3 Annotations

From a NLP point of view, an annotation links data to a text fragment or to other annotations. The data can be of many kinds, syntactic (POS, grammatical roles, etc.), semantic (lexical entry, terminological element, ontological entity, anaphora…), discursive (focus, concession, restriction, emphasis, etc.), and free comment. The text fragment can be all in one segment, or can involve several segments. In our approach as in most works focussed on technical contents, the annotations considered have constrained data (enumerated *a priori* in a closed list). The text fragments to which they are attached can be made of separate pieces, and they can be structured by other (lower level) annotations.

Different kinds of knowledge help legal texts users to search in the text, and so OMTAT uses different types of annotations to reflect the particular role of each.

Keywords (*named AnnotationK*) are exact denominations (except morphological variations) which have a specific meaning in the domain and can be by themselves a clue. For instance, *Partie* (Part), *Titre* (Title[7]), *Chapitre* (Chapter) *Article* (Article) are keywords : using a synonym of Part as *Fragment* (Fragment) is impossible, as the meaning of these words is only defined by the hierarchy in the table of contents. Note that the heading of a division is under the scope of the relevant keyword and is attached as an attribute of the annotation.

Terms (*named AnnotationT*) represent significant entities of the domain. Terms can be supported by words or multiwords, but their significance relies on the attached meaning, represented in the term-annotation label. Note that different words may receive the same meaning and that, most often, one of them is chosen to represent this meaning. In the *Code du travail*, *salaire*, *sanction* or *contrat de travail* are terms (the meaning of the first can also be supported by e.g. *traitement*, *appointements*, *paie*). As in many domains, legal terms are often gathered in specialized resources like terminologies or ontologies.

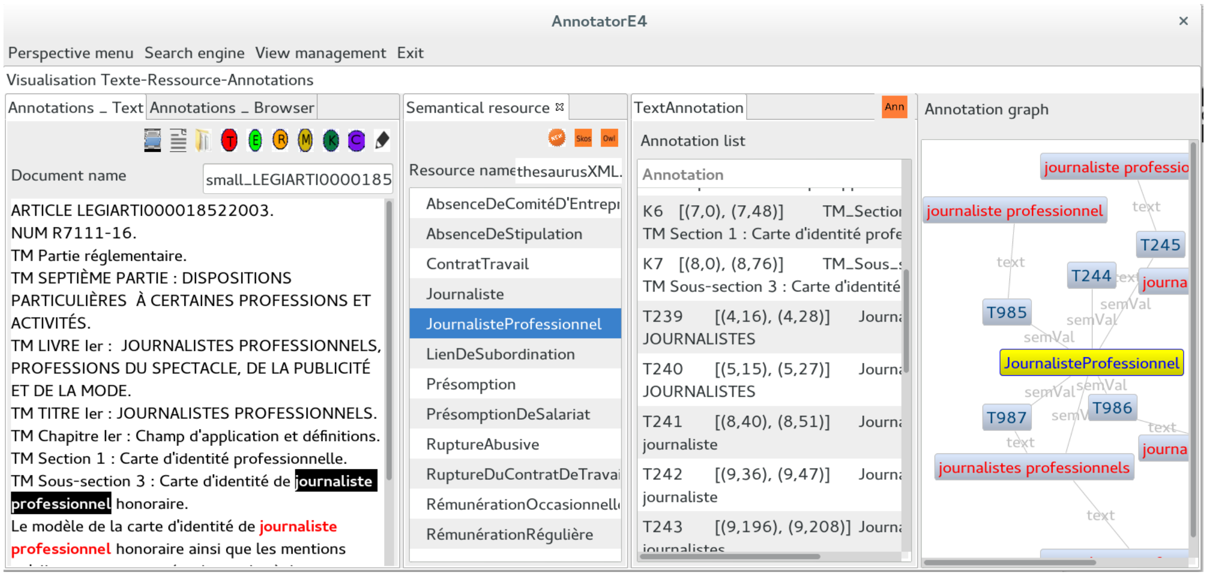
Relations (*named AnnotationR*) allow to link two terms by a specific relation. For instance, an *authority_for* relation links *entreprise de presse* to *journaliste* in *Est journaliste professionnel toute personne qui a pour activité principale, régulière et rétribuée, l'exercice de sa profession dans une ou plusieurs entreprises de presse et en tire l'essentiel de ses ressources* (Is a professional journalist any person whose main, regular and paid activity consists in exercising his profession in one or more newspaper companies and who obtains this way the main part of its resources).

References (*named AnnotationL*) annotations mark fragments which refer to other fragments of a legal text, most often with the help of a standard identifier, e.g. *La présomption de salariat prévue à **l'article L. 7121-3***, or: *Lorsque le travail du journaliste donne lieu à publication dans les conditions définies à **l'article L. 132-37 du code de la propriété intellectuelle***. When the reference is relevant for the question at hand, the text under consideration must be extended with the fragment referreed to by the annotation. Three main specific relations can link an annotation to a reference : *defined_in*, *decided_in*, *mentioned_in*

Events (*named AnnotationE*) link sets of arguments around a central predicate, the *trigger*. For instance, in *Le salaire perçu par un mannequin pour une prestation donnée* (The salary received by a model for a given performance), the trigger of the event is *perçu*, its theme is *salaire*, its agent is *un mannequin* and its cause is *une prestation donnée*.

Context (*named AnnotationC*) annotations cover a possibly large fragment having a functional

[7] In French law, Title is a level in the hierarchy of divisions, not to be mistaken for the text of a heading of any level, named its title



Document name small_LEGIARTI0000185

ARTICLE LEGIARTI000018522003.
NUM R7111-16.
TM Partie réglementaire.
TM SEPTIÈME PARTIE : DISPOSITIONS PARTICULIÈRES À CERTAINES PROFESSIONS ET ACTIVITÉS.
TM LIVRE Ier : JOURNALISTES PROFESSIONNELS, PROFESSIONS DU SPECTACLE, DE LA PUBLICITÉ ET DE LA MODE.
TM TITRE Ier : JOURNALISTES PROFESSIONNELS.
TM Chapitre Ier : Champ d'application et définitions.
TM Section 1 : Carte d'identité professionnelle.
TM Sous-section 3 : Carte d'identité de journaliste professionnel honoraire.
Le modèle de la carte d'identité de journaliste professionnel honoraire ainsi que les mentions

Resource name thesaurusXML.

AbsenceDeComitéD'Entrepr
AbsenceDeStipulation
ContratTravail
Journaliste
JournalisteProfessionnel
LienDeSubordination
Présomption
PrésomptionDeSalariat
RuptureAbusive
RuptureDuContratDeTravai
RémunérationOccasionnell
RémunérationRégulière

Annotation
K6 [(7,0), (7,48)] TM_Sectio
TM Section 1 : Carte d'identité profe
K7 [(8,0), (8,76)] TM_Sous_s
TM Sous-section 3 : Carte d'identité
T239 [(4,16), (4,28)] Journa
JOURNALISTES
T240 [(5,15), (5,27)] Journa
JOURNALISTES
T241 [(8,40), (8,51)] Journa
journaliste
T242 [(9,36), (9,47)] Journa
journaliste
T243 [(9,196), (9,208)] Journa
journalistes

Figure 1: Main annotation view

role by which interpretation has some specificity. In the *Code du Travail*, context annotations correspond to entities in the table of contents, because the functional view is reflected in this table. For instance, article L7111-3 states *Si l'employeur est à l'initiative de la rupture, le salarié a droit à une indemnité qui ne peut être inférieure à la somme représentant un mois, par année ou fraction d'année de collaboration, des derniers appointements*[8]. It cannot be accurately understood without accounting for its position: in the legislative part, part VII (Rules specific to some professions), Livre 1 Title 1 (journalists), Chapter 2 (employment contract) Section 2 (breach of contract) - hence "the salaryman" need to be a professional journalist. Note that, in other texts as case law decisions for instance, judgments have functional parts which remain implicit, while they must nevertheless be recognized.

Terms, relations and events have been widely used in the BioNLP challenge, thanks to the outer encoding provided by the Brat tool[9] ((Stenetorp et al., 2012b; Stenetorp et al., 2012a)). Other annotations use the same kind of encoding and have been devised for the specific needs of legal annotation.

4 OMTAT Tool

In this section, we present the OMTAT (One More Text Annotation Tool) system, built as an E4 Eclipse application to implement annotations as described above. Many functionalities have been defined to create and/or visualize different annotations. The main window involves four views (see fig 1):

- Text view: it shows the text of the document (and its name). All the annotations are emphasized with different text colors according to their type.

- Semantic resource view: it shows the semantic resource (a thesaurus or an ontology). Any semantic resource in SKOS format or OWL format may be loaded. A SKOS semantic resource may be enriched when an annotationT is created and its semantic value does not exist in the resource. A OWL semantic resource cannot be modified. In the figure, the semantic resource is a thesaurus. It is built on the labour code and restricted to the use case.

- Text annotation view: it displays all the annotations defined on the document showed in the text view. A click on an annotation selects the corresponding text in the text view.

[8] Translation: If the breach is initiated by the employer, the salaryman has a right to an indemnity which cannot be less than the amount of money representing one month per year of fragment of a year of collaboration, of the last salary

[9] http://brat.nlplab.org/about.html In the Brat tradition, Terms are named *Types*

- Annotation graph view: it shows the selected annotations in the form of a graph which nodes correspond to values and edges correspond to labels.

Other global functionalities include:

- Text and sets of annotationK(s) may be extracted from XML files provided that markups are defined in a configuration file.

- A semantic resource provided with preferred labels may be cast on a text to create terminological annotations (AnnotationT).

From the annotation point of view, OMTAT has common points with well known tools, GATE [10] (Cunningham, 2002; Cunningham et al., 2011) and BRAT[11] (Stenetorp et al., 2012b), but has also significant differences. GATE considers one single category of annotations of the form (type, span, features) ; the type is a single word, features are key-value pairs. Gate can model structured annotations with the help of a “constituent” feature which records lists. GATE documentation states that “No special operations are provided in the current architecture for manipulating constituents” [12]. After version 7, the provided library allows modeling relations through a `members[]` array[13]. The 8.1 basic version does not offer a user interface to manually add relations as it does for plain annotations. A search tool locates the next occurrence of a string or regex and can restrict the search to the scope of existing annotations.

Brat has three categories of annotations: types, relations and events. *Types* are the basic blocks associating a (possibly discontinuous) portion of text with a type. A *Relation* involves two typed annotations and a labelled link. An *Event* is made of a main typed annotation (its *trigger*) linked to a variable number of arguments. The last two can be represented in Gate by relations, but Brat emphasizes their difference from a user point of view: the relation need no lexical support (e.g. there is no lexical link between a bacterium and the mouth which is its localization) while the trigger of an event is its lexical support and restricts possible arguments. Last, Brat is designed to be used as a centralized collaborative tool through a web server. Its user interface combines text and drawings to visualize or add all categories of annotations, while the annotation schema is centrally controlled. A search tool locates annotations of a given category according to the conjunction of conditions on their text and on their attributes.

In comparison, OMTAT is a single user tool under Java as Gate (but Gate has a collaborative extension). It has the three categories of Bratt, it can read its standoff format and defines some more categories to account for the structure of legal documents. It accepts to dynamically manage semantic resources (Gate also does) and includes a search engine which is described in the following section.

5 The Search Engine

An experimental search engine has been developed for the need of exploring the annotated corpus currently in memory. Its main role is to build so called w-tuples, tuples of elements (i.e. annotations, sentences and documents) constrained by a set of conditions. At the engine level, a query has a simple `Select` ... `From` ... `Where` ... form and returns s-tuples, tuples of attribute values. For example, figure 2 shows a plain graphic interface to the engine and a query returning the list of pairs $(S.text, Aart.text)$ where $S.text$ is the text of a sentence containing a Term annotation *Presumption of Salary*, and $Aart.text$ is the number of the article containing the sentence[14]. The function of each clause is shortly explained here:

- The `From` clause provides a set of open (in memory) documents in which the search will happen.

- The `Where` clause describes the form of w-tuples and the conditions that they must satisfy. For that purpose, every element in the tuple is named and typed (the type may be annotation, sentence or document; for the sake of conciseness, the first letter of the name involves the type). Conditions are then relations between attributes of the elements and possibly constant values. For example, $S.numsent < 5$ is a condition requiring that

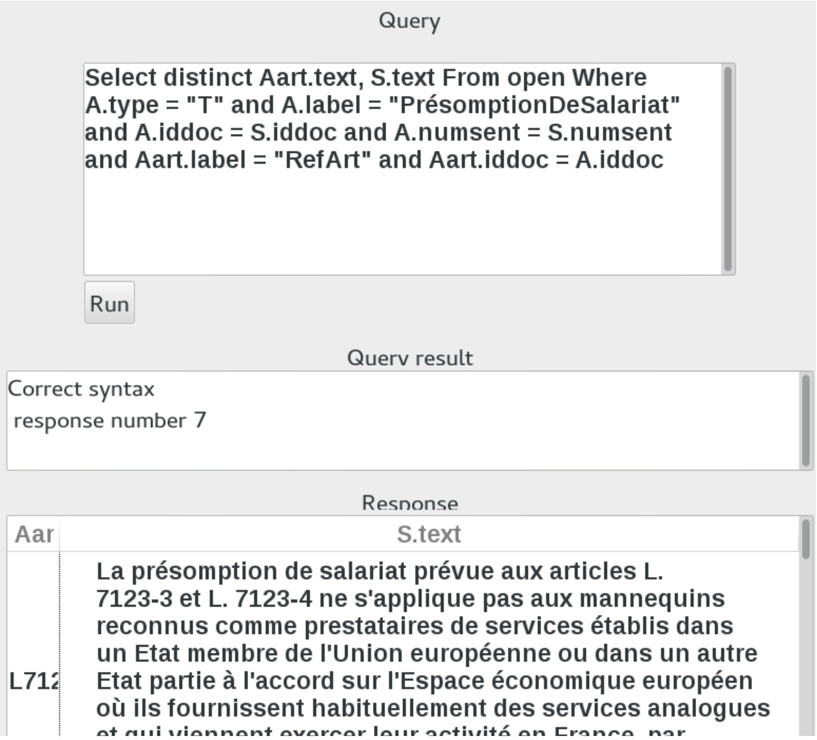
[10] General Architecture and Text Engineering, http://gate.ac.uk

[11] Brat rapid annotation tool, http://brat.nlplab.org/

[12] Gate inline documentation, section 5.4.2

[13] Documentation, section 7.7

[14] Hovering the mouse over truncated sentences shows the full text. The OK button saves the result.



```
Select distinct Aart.text, S.text From open Where
A.type = "T" and A.label = "PrésomptionDeSalariat"
and A.iddoc = S.iddoc and A.numsent = S.numsent
and Aart.label = "RefArt" and Aart.iddoc = A.iddoc
```

```
Correct syntax
 response number 7
```

Aar	S.text
L712	La présomption de salariat prévue aux articles L. 7123-3 et L. 7123-4 ne s'applique pas aux mannequins reconnus comme prestataires de services établis dans un Etat membre de l'Union européenne ou dans un autre Etat partie à l'accord sur l'Espace économique européen où ils fournissent habituellement des services analogues et qui viennent exercer leur activité en France, par

Figure 2: The Query view

the element S (a sentence) be at the beginning of the document (in the first five sentences), and $A\_obj.label = A\_agent.label$ is a condition requiring that the two elements A_obj and A_agent (both annotations) share the same label.

- The `Select` clause describes what s-tuple is returned for each built w-tuple. Only *element.attribute* names can be used here, and all the element names must be defined in the w-tuples, i.e. used in the `Where` clause. Any attribute of a defined name can be used in the Select. Last, a `Distinct` modifier allows to merge identical returned tuples.

For instance, the query

```
Select distinct A1.numsent
From current
Where A1.numsent = A2.numsent
   and A1.idannot != A2.idannot
   and A1.label = A2.label
```

returns the sentence number of those sentences in the current document in which the same label occurs twice.

The different kinds of annotations described in section 3 are implemented as classes, benefiting from inheritance. Attributes are provided by a specific inner mechanism of annotation classes allowing to define computed attributes which are not explicit in the Brat format (the size of the text, the number of arguments of an event, etc.). The kind of annotations is identified by the $type$ attribute. Some more attributes are common to all annotations, and others are specific to one or several types. In the present version, common attributes are the identifier of the annotation, its type, its label, its size, its start and end position, its sentence number and the identifier of its document. Other attributes may be for instance the subject and the object of a relation (type "R") or the division of a context annotation (type "C"), i.e. the kind of textual unit (chapter, paragraph) which defines its borders. Elementary conditions in the `Where` clause are only combined by conjunction; a limited form of disjunction is available through inequalities and string relations *starts with*, *ends with*.

Parsing provides a control of the query and of its conditions. Elementary conditions are sorted according to the names in their left and right hand part. w-tuples are then built recursively on the set of names involved. In the application of conditions, the validity of many attributes can only be decided when the type is known; furthermore, some of them are optional in the type (*e.g.* the arguments of an event). So the validity of attributes used by conditions is dynamically determined while the w-tuples are built.

6 Use Case

6.1 Presentation

We have considered annotation of the French *Code du Travail*, which has 4644 articles. In a legal sense, a French Code groups and organizes the legal and regulatory texts produced along time for a given domain. Jurisprudential knowledge comes from other sources. A legal user of this text must discover as quickly and easily as possible which fragments are relevant to his problem, and may have to examine many excerpts.

Following a well established layout of French codes, we stored each article in a separate document which includes a reminder of its position in the table of content: the document starts with headings of all levels in the scope of which it is. At the moment, a small set of annotations have been marked in the text: *Contrat de travail* (employment contract), *Journaliste* (journalist), *Journaliste professionnel* (professional journalist), *Présomption* (presumption), *Rémunération* (remuneration).

Recall the example legal problem of section 1, for which our approach may help a NLEP to find useful excerpts of the labour code answering to his problem. A freelance journalist used to work for a newspaper, and at a moment the newspaper does not ask him anymore article. Being freelance, the journalist is paid for each provided product (article, photograph, drawing, etc..). In the case, he has never signed any written contract whatsoever. The stakes are if he has a right to an indemnity.

6.2 A first Analysis

Articles can be selected according to the presence of annotations *Journaliste* in the document, including titles. This yield 336 distinct answers, due in particular to one of the titles of level *Livre*, which enumerates various professions, so for instance articles about (theater, movies, …)-players are also under this title. Restricting the annotation to occur in the body of the article reduces to 45 articles, 17 of which are in laws and can be focused on by requiring a K annotation with a *Partie Législative* label. Requiring a supplementary annotation on *Contrat de Travail* only leaves 5 articles.

Among these, article L 7112-1 states *Toute convention par laquelle une entreprise de presse s'assure, moyennant rémunération, le concours d'un journaliste professionnel est présumée être un contrat de travail. Cette présomption subsiste quels que soient le mode et le montant de la rémunération ainsi que la qualification donnée à la convention par les parties*[15]. Note also that the section is entitled *Présomption de salariat* (Presumption of wage relation), implying that wages are a form of payement specifically attached to employment contracts.

This means that, provided he is a professional, the freelance is presumed to be governed by an employment contract and hence to benefit of a right to an indemnity. The user must enjoy a minimal understanding of legal reasoning to supplement this basic information with the help of some more queries. Namely, he must know that, beside employment, other kinds of contracts may be relevant in order to "remunerate the support" of some individual, and that employment contracts themselves may have different categories. Searching for *Journaliste* and *Remuneration* annotations in the same article gives six results,

One of them is L. 7113-3: *Lorsque le travail du journaliste professionnel donne lieu à publication dans les conditions définies à l'article L. 132-37 du code de la propriété intellectuelle, la rémunération qu'il perçoit est un salaire*[16]. It means that it is possible to argue that the payment is of the kind specific to intellectual property, i.e. an authorship rights remuneration, and not an employment contract. The conditions of this possibility are to be found in the Intellectual Property Code. To maintain the legal complexity inside reasonable bounds, this path is not followed here.

[15] Any agreement by which a press company obtains, through remuneration, the support of a professional journalist, is presumed to be an employment contract. This presumption remains whatever can be the mode and the amount of the remuneration or how participants qualify the agreement.

[16] When the work of the professional journalist gives rise to a publication in such conditions as defined in article L. 132-37 of the Intellectual Property Code, the remuneration that he receives is a salary

6.3 More on the Contract

Every French worker knows that employment contracts belong to one of two categories : contracts with an indeterminate duration (CDI) or with a determinate one (CDD), and that they do not involve the same rights. More, article L. 7112-2, one of the five obtained from the first query, considers a breach of the CDI, while no mention is made of a CDD. Definitions can only be found considering articles which apply to the generic case. Searching for titles of the highest possible level annotated with *employment contract* yields Part 1 (individual employment relations) Livre II (Employment contract).

Searching in this *Livre* for articles annotated both with CDI and CDD provides two basic texts. In L. 1221-2 it is stated that the CDI is the default case: *Le contrat de travail à durée indéterminée est la forme normale et générale de la relation de travail. Toutefois, le contrat de travail peut comporter un terme fixé avec précision dès sa conclusion ou résultant de la réalisation de l'objet pour lequel il est conclu dans les cas et dans les conditions mentionnés au titre IV relatif au contrat de travail à durée determinée*[17]. And in L. 1242-12, this is enforced by requirements on the form of the CDD: *Le contrat de travail à durée déterminée est établi par écrit et comporte la définition précise de son motif. A défaut, il est réputé conclu pour une durée indéterminée*[18].

7 Conclusion

This article has described the use of different textual annotations to help legal documents search take advantage of the document structure. An experimental use case demonstrates the utility of the approach. An implementation has been carried out which allows to explore, query and add annotations. Future work will allow us to deepen the study of links between semantic annotations and semantic document search.

[17] The CDI is the normal and standard form of employment relation. However, the employment contract may involve an end date precisely fixed at start or determined by the achievement of the object in view of which it is decided, in cases and conditions mentioned in Titre IV related to CDD

[18] The CDD is set up in writing and contains a precise definition of its motive. Failing that, it is deemed to be agreed for an indeterminate duration

Thanks

We thank Jorge Garcia Flores and Nadi Tomeh for their friendly review of our writings.

References

Michael W. Berry, R. Esau, and Bruce Keifer. 2012. The use of text mining techniques in electronic discovery for legal matters. In C. Jouis, I. Biskri, Jean-Gabriel Ganascia, and M. Roux, editors, *Next Generation Search Engines: Advanced Models for Information Retrieval*, pages 174–190. IGI Global.

A. Boer, A. Winkels, and F. Vitali, 2008. *Metalex xml and the legal knowledge interchange format.*, volume Computable Models of the Law - Lecture Notes in Computer Science -4884, pages 21–41. Springer.

Hamish Cunningham, Diana Maynard, Kalina Bontcheva, Valentin Tablan, Niraj Aswani, Ian Roberts, Genevieve Gorrell, Adam Funk, Angus Roberts, Danica Damljanovic, Thomas Heitz, Mark A. Greenwood, Horacio Saggion, Johann Petrak, Yaoyong Li, and Wim Peters. 2011. *Text Processing with GATE (Version 6).*

H. Cunningham. 2002. Gate - a general architecture for text engineering. In *Computers and the Humanities, Volume 36*, pages 223–254.

Nada Mimouni. 2015. *Interrogation d'un réseau sémantique de documents: l'intertextualité dans l'accès à l'information juridique*. Ph.D. thesis, Paris 13 - Sorbonne Paris Cité, janvier.

M. Palmirani, R. Brighi, and M. Massini. 2005. Automated extraction of normative references in legal texts. In NY USA ACM, editor, *Proccedings of the 9th international conference on Artificial intelligence and Law, ICAIL*, pages 105–106.

Pontus Stenetorp, Sampo Pyysalo, Goran Topić, Sophia Ananiadou, and Akiko Aizawa. 2012a. Normalisation with the BRAT rapid annotation tool. In *Proceedings of the 5th International Symposium on Semantic Mining in Biomedicine*, Zürich, Switzerland, September.

Pontus Stenetorp, Sampo Pyysalo, Goran Topić, Tomoko Ohta, Sophia Ananiadou, and Jun'ichi Tsujii. 2012b. 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*, pages 102–107, Avignon, France, April. Association for Computational Linguistics.