Traitement automatique des langues

Traitement automatique de la langue juridique / Legal Natural Language Processing

sous la direction de
Adeline Nazarenko
Adam Wyner

Vol. 58 - n°2 / 2017

ATALA

Traitement automatique de la langue juridique / Legal Natural Language Processing

ATALA

Revue de l'Association pour le Traitement Automatique des Langues

TAL
Vol.
58

n°2
2017

Traitement automatique de la langue juridique / Legal Natural Language Processing

ATALA

Traitement automatique des langues

Revue publiée depuis 1960 par l'Association pour le Traitement Automatique des LAngues (ATALA), avec le concours du CNRS, de l'Université Paris VII et de l'Université de Provence

ISSN 1965-0906

https ://www.atala.org/revuetal

Traitement automatique des langues

Secrétaire
Aurélie Névéol - LIMSI, CNRS

Traitement automatique des langues

Volume 58 – n°2 / 2017

TRAITEMENT AUTOMATIQUE DE LA LANGUE JURIDIQUE / LEGAL NATURAL LANGUAGE PROCESSING

Table des matières

Legal NLP Introduction

Adeline Nazarenko* — **Adam Wyner****

* *LIPN, Université Paris 13 – Sorbonne Paris Cité& CNRS*
** *Swansea University, School of Law and Department of Computer Science*

ABSTRACT. *Language and law have always had a close relationship, the latter being primarily a "discourse". Currently, the automatic processing of the legal language is a major issue because of the growing footprint of the law on the web and its complexity in contemporary globalised societies. In addition, through the prism of a specialised language, here legal language, we can measure the progress of natural language processing (NLP). The goal is to integrate different processes into operational applications that meet specific needs. This goal is particularly challenging and important in the case of legal language due to the intertwined levels of linguistic analysis, from the analysis of character strings (for the identification of citations, for example) to argumentation. This issue of TAL aims to draw attention to the issues and challenges of legal NLP, to present recent research in this field, and, more broadly, to show how different methods of analysis are organised for this specialised language.*

RÉSUMÉ. *La langue et le droit entretiennent depuis toujours des liens étroits, ce dernier étant d'abord un « discours », mais le traitement automatique de la langue juridique représente aujourd'hui un enjeu majeur du fait de l'empreinte croissante du droit sur le web, de son ouverture et de sa complexification dans les sociétés contemporaines mondialisées. Par ailleurs, le prisme d'une langue de spécialité, ici la langue juridique, permet de mesurer les progrès du traitement automatique des langues. Il s'agit d'intégrer différents traitements dans des applications opérationnelles qui répondent à des besoins spécifiques, et ce défi de l'intégration est particulièrement important dans le cas de la langue juridique du fait de l'intrication des paliers d'analyse linguistique, depuis l'analyse des chaînes de caractères (pour le repérage des citations, par exemple) jusqu'à celle de l'argumentation. Ce numéro de TAL vise à attirer l'attention sur les défis et les enjeux du traitement automatique de la langue juridique, à montrer l'intérêt des recherches récentes dans ce domaine, mais aussi, plus largement, à montrer comment différents méthodes d'analyse s'organisent pour une langue de spécialité.*

KEYWORDS: *Law, LegalTech applications, processing of languages for special purposes.*

MOTS-CLÉS : *droit, applications juridiques, traitement des langues de spécialité.*

1. Introduction

The aim of this issue of the *TAL journal*, entitled "Automatic Processing of Legal Language", is to contribute to the analysis and exploitation of legal information using Natural Language Processing (NLP), considering theoretical problems and specific linguistic phenomena as well as how the analyses are integrated or applied to processing legal sources. It highlights work currently being done in the field, stimulates new research strands, and shows more broadly how different methods of analysis can be organised for processing special language.

Language and Law have always been closely linked, the latter being fundamentally a "discourse". Legal language is to be understood here in a broad sense: written and oral language, legal texts and judgments, as well as regulatory texts such as decrees, regulations, contracts or requirements. As Language and Law are inseparable, it is essential to develop methods of Legal NLP in order to understand legal language and discourse, to develop tools supporting the exploitation of legal sources for law enforcement, as well as to enable transparent, international, and interoperable legal systems over the web.

This introduction presents a selection of challenges that legal language processing faces today; as we are not offering a complete review of the field, there are naturally other research challenges and approaches which could be discussed [1]. Nonetheless, our brief survey shows that NLP is a key issue in the development of LegalTech (Section 2), that legal language raises specific difficulties in terms of NLP (Section 3), and that integration remains a major problem when NLP technologies are to be used in domain-specific but also in real-world applications (Section 4). Section 5 also introduces the papers that compose this special issue and illustrate various facets of the challenges that current research in NLP is addressing.

2. LegalTech – context and opportunities

Broadly, LegalTech refers to technologies from Computer Science that are applied to a range of areas related to legal practice and materials. In order to set the context and opportunities, we scope the discussion. LegalTech has a wide range of application areas to help law firms and organisations with daily activities related to document support (creation, revision, storage and retrieval), legal proceedings (providing electronic documents in the course of litigation and government investigations, legal research of (non-)legal sources to support decision-making) and more generally all aspects related to the dematerialisation of legal services from text and paper to digital form. Some of these areas are document and language centered (document storage and retrieval, electronic discovery, legal research, and document automation/assembly) and highly relevant to Legal NLP.

1. See recent editions of the JURIX and ICAIL conferences as well as issues of the *Journal of Artificial Intelligence and Law*.

Another well-developed area of research and development has been Forensic Linguistics, which applies linguistic analysis to legal materials in proceedings (Osslon, 2009; Coulthard and Johnson, 2010). Amongst the many topics in Forensic Linguistics, we can find linguistic analyses of discourses by participants in police investigations and interviews, courtroom exchanges, authorship identification along with associated opinions, security analysis, and translation of documents in multilingual legal contexts. [2] To a greater or lesser extent, research in Forensic Linguistics has applied NLP technologies, though this is not essential to the endeavours, which can often be carried out by manually annotating or marking up text.

Language processing has, then, been central to address tasks and purposes in areas of LegalTech and Forensic Linguistics. To the extent that these tasks and purposes have solutions and are already commercialised, we can say they are applications of existing technologies to well defined and scoped textual issues. We find large, well-established legal information service providers such as Thomson Reuters and Oracle, law firms such as Pinset Masons and Riverview Law, as well as a host of startups touching on a full spectrum of issues. [3, 4] However, for the NLP research community, the aim is to take up opportunities in identifying and addressing challenging textual issues. We mention some of them.

NLP technologies were first used for assisting the drafting of legal documents. One common approach to automated support relies on a decision-tree model of drafting, where a document template (e.g. a contract) is automatically refined and instantiated according to the drafter's local decisions (Sprowl, 1980; Gordon, 1989). Such approaches use quite basic NLP technology to provide contracts. However, NLP now has significant opportunities in the analysis of legal documents, which have become available in very large scale, for example, enabling the mining of contractual relationships across a corpus of documents, e.g. global oil and gas concessions.

Traditionally, law schools, legal offices, and legislative counsels have produced guidelines to explain what should be the internal structure and constituents of legal documents (*e.g.* resolutions, executive orders, contracts, regulations), how to express rules and decisions unambiguously using precise legal terminology, and how to handle cross-referencing between sources, etc. In this context, NLP is naturally used to support drafting by controlling the structure of the documents, the length of the sentences, and the use of recommended terms (Höfler, 2012).

A more complex issue is related to the control of legality and consistency of legal sources. As anyone knows, legal documents can be very long, they are frequently updated, and they are part of large legal systems which are subject to interpretation and evolve with various social and political factors. Legal actors must ensure a consistent and up-to-date use of terminology, they must control the compatibility and the consistent evolution of rules that come from different jurisdictions, and they must check

2. See `http://www.iafl.org/` for relevant conferences and journals.
3. `https://angel.co/legal-tech-1`
4. `https://www.legalgeek.co/startup-map/`

the legality of the sources and decisions with respect to constitutional norm. Beyond surface analysis, these controls involve a deep understanding of the legal texts and logical reasoning. The development of standards for encoding the structure of documents (e.g. *LegalDocML*[5]) and the semantic content of rules [e.g. *LegalRuleML* (Athan *et al.*, 2015)] is a prerequisite for the construction of tools to facilitate control.

Retrieving information is also a major challenge, considering the huge bulk of laws and decisions that are produced over the years in modern societies. They cover economic, social and political issues on the local scale as well as worldwide. For example, citizens want to know which rules apply to their district when they want to restore their houses; employers must know the applicable labor legislation; and trade is governed by international treaties and agreements. One must be able to retrieve the relevant texts and to extract the specific legal and statutory rules that are relevant to a given case. Most countries have an official website for the legislation[6], regulations, and legal information. However, ensuring the publication, interoperability, and accessibility of these resources calls for advanced semantic and search technologies. There are needs for richer metadata (e.g. date, jurisdiction, legal matter, keywords, etc.) but also fine-grained search or navigation tools. For instance, one should be able to directly find the decisions related to a given topic that derive from the transposition of a given European directive (Mimouni *et al.*, 2014).

Related to information retrieval is linking legal resources. In their daily work, legal professionals, such as barristers, judges, prosecutors, legal advisers, analyse the law and the cases, search for precedent cases, identify relevant legislative or regulatory documents, and consider jurisprudence. In other words, a range of resources must be leveraged to gain a "wholistic" view of the matter at hand. Thus, there must be a *legal semantic web*, where sources are accessible, interconnected, annotated such that legal professionals can query, explore, and possibly enrich corpora with new cases and documents with new interpretations (Casanovas *et al.*, 2016).

Information retrieval and linking serve to support decision-making, reasoning, and compliance. Taking into consideration contracts, legislation, and regulations, legal professionals ought to be able to determine whether a given action is legally compliant or secure, what legal determination follows from given input information, or what one's liabilities are. Without such capabilities, business, government, and individual activities can be unclear or problematic. For example, one cannot manage multimedia content without mastering the various rights attached to the elements, where the rights are often encoded in distinct contracts. One would want to automatically compare contracts. To ensure legal certainty, one needs tools to analyse the legal documents, extract and formalise the rules attached to each type of content, reason over the set of rules that apply to a specific business process, and test that the action abides by compliance or security protocols.

5. See `https://www.oasis-open.org/committees/tc_home.php?wg_abbrev=legaldocml`, which subsumes *Akoma Ntoso* `http://www.akomantoso.org/`

6. e.g. `https://www.legifrance.gouv.fr/` or `http://www.legislation.gov.uk/`

In addition to reasoning with respect to a given body of legal documents, disputes about the interpretation or application of the law arise. Argumentation is often at the heart of legal reasoning and judgment: one wants to validate the arguments and claims put forward as well as to suggest new pro and con arguments. Supporting argumentation calls for NLP technologies to mine, collect, and interrelate arguments in legal sources, to populate knowledge representation of argumentation models, and to reason over the models (Wyner *et al.*, 2010). The ultimate goal is to support legal professionals in designing, understanding and controlling arguments and more generally to assist legal reasoning.

The various topic areas above (drafting, managing, retrieving, linking, reasoning, compliance, and so on) apply as much across jurisdictions as within. Thus, there are also needs for tools to support comparative law, which is crucial for working with the law in international and global settings. Here again, we need to understand the texts in depth, to extract and model the rules they contain, then to reason on those rules to check their consistency and identify flaws or redundancies. Moreover, these operations must be applied on a large scale and with respect to different legal systems, where terminology and concepts must be correlated or aligned between jurisdictions.

In sum, drafting, publishing, querying, linking, and reasoning over legal sources in ways that can be exploited by legal practitioners as well as by citizens and governments raise significant challenges that relate to NLP in connection with document engineering, information retrieval, knowledge representation and reasoning, as well as more generally decision support.

3. A challenging field for NLP

NLP is itself a mature sub-area of Artificial Intelligence with a range of well-known rule-based or machine-learning techniques along with tools designed to split sentences, tokenise texts, lemmatise words, tag words with part-of-speech, parse sentences, enrich text with semantic roles, recognise named entities, extract relations between entities, identify discourse markers, perform anaphoric reference, classify texts or textual passages, and so on. For certain classes of text, e.g. newspaper and narrative text, such techniques and tools are often highly successful.

However, such techniques and tools are usually developed on and for specific textual types. Yet, what works for newspaper or narrative texts does not succeed for corpora of legal language. The challenges are not so much the volume of texts, but 1) the technicalities needed to account for particular, complex structures and patterns and 2) the specificity of performance requirements. We discuss these below.

3.1. *Complexity of legal sources*

Legal sources are complex in many ways. We highlight some of them.

The language itself is complex. To start with, the lexicon is rich. It includes uniquely legal expressions (e.g. *charged* or *defendant*) some of which may have internal structure (e.g. *without prejudice to any claim*), which are mostly standardised or codified in dictionaries of legal language. In addition, there are legal sub-areas, e.g. *family*, *criminal*, and *contract* law, each of which codify different terminologies. Both the legally-specific lexical items within and across legal sub-areas are important for the processing of legal documents. Moreover, in at least the common law context, criminal law, tort law, civil responsibilities, and to some extent legislation and regulations embed ordinary language or descriptions, which must also be analysed. As legal concepts bear on everyday occurrences, such as parking restrictions or temporal relations in a murder, legal and non-legal expressions must correctly be tied together. Considering the variation in time, jurisdictions, and languages, the lexical complexities proliferate.

In terms of syntax, sentences are generally well constructed (as opposed to "noisy" texts published on social media, for instance) but often long and highly structured, with embedded or intertwined clauses and sometimes ambiguities (*e.g.* prepositional clause attachment, scope of an adverb). The presentational format may break a long sentence into parts that are distributed in lists, *e.g.* "A British citizen is a person: (a) born in the UK; (b) born to a British citizen; ...", which themselves may contain sublists. Specific NLP modules must be developed for analysing the various form of lists that can be found in corpora. In practice, it can be observed that the current parsers, e.g. the Stanford Parser, which have been trained on news corpora, can fail to parse long sentences and to resolve ambiguities of legal texts.

This complexity is also found at the semantic level. Anaphora resolution is problematic given the length and complexity of sentences. Logical translations, while largely feasible for short sentences of non-legal language, are infeasible with legal texts: it is difficult to determine the scope of the logical operators (negation, conjunction, disjunction, and conditional) with respect to noun phrases and verb phrases; similarly, the scope of quantifiers can be problematic. These issues are not specific to legal language, but their resolution is of the utmost importance in the law field, because they are prominent in legal sources and because legal reasoning is based on logical analysis and the interpretation of those sources.

Stylistics also matters. There are few standards that hold across international jurisdictions for how legislation or regulations are expressed other than those standards specifically set by international organisations. With respect to common law contexts, case reports are largely up to the presentational preferences and styles of the judges and clerks who write up the report. NLP tools must confront the absence of widespread, consistent, and homogeneous linguistic expression.

Beyond the language of individual texts, the corpora themselves are complex because they are very structured and closely interconnected. An article of law often cannot be interpreted in isolation. It must be considered in the light of the law of which it is a part and of all the texts which are attached to it, through a term definition, by references and citations, or by semantic relations. For instance, when the enforcement of a given law t_2, which transposes the European directive t_1, is suspended upon the adoption of a decree t_3 that specifies its conditions of application, none of the three texts can be taken independently of the others. In case law, cases are semantically linked in relations of *upholds*, *overrules*, and similar concepts (so-called Shepardisation). Finally, legal texts are not immutable, but can be wholly or partially updated, leading to portions of the text referring to updated or overwritten law. Several versions of the same law article usually coexist, each one being valid for a specific time frame.

In the various ways outlined above, legal texts are complex, rich, and diverse, making them challenges to current NLP tools.

3.2. *Performance requirements for legal applications*

Another challenge for processing legal texts is related to the performance requirements of the legal system. Such requirements may vary from one application or use-case to another, and they may be especially high in legal NLP.

In legal case-based reasoning, for instance, it is important to carry out research to identify relevant precedent cases or applicable legislation. Here, legal information service providers have large teams of legal experts to analyse and index legal texts or to provide summaries of the contents of legal texts (also known as headnotes), providing meta-data that can be used in searches. Nonetheless, legal professionals widely accept the results of querying such resources, which can then be subjected to further manual filtering. Were open NLP techniques to be applied to the source texts, the results ought to at least mirror those provided by human experts. The requirements on getting the correct textual resources may be very high, given that a legal argument is only as well supported and defended as the volume and accuracy of the material used. In principle, failing to adequately defend against even a single precedent case or to take a piece of regulation into account could be fatal.

In the domain of the translation of legal texts, given that the language of the law is paramount, the quality of the translation ought to be not only very high, but ought to be validated in some systematic and transparent manner. After all, an obscure phrase in either the source or translated text could have legal ramifications should the phrase be the lynchpin of the dispute.

Even where precision and recall were hypothetically perfect, the results may not satisfy the requirements of the legal problem. Suppose we have a very large corpus of death penalty decisions split between verdicts of innocence or guilt. Suppose a machine learning classifier has outstanding performance in classifying the cases. From this, one might suppose that given all the information required as input to a particular

novel case (on a par with that provided to cases in the case base), the system classifies the verdict, *e.g.* as guilty. Should the defendant then be accepted as guilty? It is highly unlikely that the legal system much less the defendant would accept this as a definitive result. The law has long functioned by providing a full explanation to justify the decision. In part, this is to ensure that the application of the law is based on precedent and existing legislation in a clear, transparent, and systematic fashion; and in part, such decisions can be reused in a variety of ways, *e.g.* linking cases, setting precedent, and as the basis for legal appeals. Without a fine-grained representation of the internal structure of the decision, explanation to the requisite degree would be unavailable. Given the current technology, machine learning does not provide such *explanations* or structured information. This highlights that the NLP techniques ought to serve the purposes of the law and legal setting as they are.

It is not just about finding documents, making a decision, and explaining it, but often users need contextual elements, syntheses, and methodological guides. Consider a citizen accessing some online legal advisory facility using question-answering. She may want not only a document relevant to addressing a legal issue, but auxiliary information as well as guidance on how to work through towards resolution of the matter. Given the complexities of legal sources noted above, it would be unrealistic to simply link the citizen to the source material. Nor can it reasonably be expected that, from the textual sources, the citizen can understand how to navigate the process. The requirements are that the citizen needs some exactly relevant (*i.e.* high precision and recall) and clear digest of the legal materials (*e.g.* a summarisation) as well as a structured path through the legal procedures.

In legal applications, users expect very high performance in terms of results, explanations of the results, guidance on the law relevant to the results, all of which ought to be provided by easy to use tools.

4. The challenge of integration

Focusing on a special language, we can measure the progress of natural language processing. Here we focus on the language of the law, though similar issues arise for any language for specific purposes. In addition, there must be a strong focus on the integration across linguistic analyses, from the character string level (to identify for instance the citations and the list structures), to the variety of standard NLP tasks (parsing, anaphora, ambiguity resolution, etc.), to the multilingual, and to knowledge levels that support argumentation and reasoning over various legal systems. In the most advanced applications, all these various levels of analysis must be addressed and combined.

When one is interested in processing a specialised language, one necessarily has a transversal vision, which covers all levels of analysis. This is particularly true in the case of legal language, which raises great questions of understanding and reasoning:

– document engineering is a requirement. Processing of legal sources presupposes document normalisation and a standard for encoding those documents. This issue has been identified for a long time. Important standard proposals have been made for encoding the structure of legal documents, *e.g.* Akoma Ntoso [7] (Casanovas *et al.*, 2016) or LegalDocML (Athan *et al.*, 2015);

– syntactic analysis is a challenge due to the complexity and precision of legal language but also to the presence of ambiguities. Both statistical and logic-based approaches have been tested with mixed results (Wyner and Peters, 2011; Dragoni *et al.*, 2016);

– terminology is an issue in any specialised language but legal terms and idioms (*e.g. établissement d'utilité publique*), which are often complex and difficult to understand for the lay man, have a strong semantics; they are often the keystone of legal reasoning; In legal texts, legal terminology is often mixed with the vocabulary of the field (domain terminology) covered by the law, which can itself be complex. For instance, a legal text about cyberlaw will discuss how legal concepts, e.g. obligations or rights, apply to specific domain terminology for aspects of computers and communication technologies, e.g. Wi-Fi or passwords, that are neither legally defined, nor legal concepts. This is an additional difficulty for processing legal sources (Bonin *et al.*, 2010);

– stylistics is important, if we consider that drafting guidelines aims to reduce ambiguities and ease the reading of legal documents;

– semantic analysis – be it shallow or deep – is at the heart of legal content management, from information retrieval to consistency checking, and of legal reasoning. Various subtasks, such as semantic annotation (Francesconi, 2016) or rule extraction (Dragoni *et al.*, 2016), have been addressed, but developing a generic legal semantic parser for legal sources remains an open research issue [8];

– discourse analysis helps to organise and contextualise legal contents. In particular, the analysis of document networks has attracted a lot of attention (Winkels and de Ruyter, 2011; Boulet *et al.*, 2011; Christensen *et al.*, 2016);

– argumentation being the basis of legal reasoning, it is essential to rely on NLP to help produce, check and mine arguments. The analysis of legal arguments has been a topic for some time (Moens *et al.*, 2007);

– beyond NLP, knowledge engineering is also required to design semantic resources or ontologies (Sartor *et al.*, 2013) which can be used to ground semantic analysis, to formalise and apply the legal rules that cannot be exploited in their natural language form, and to enable temporal reasoning in a field where various document timelines are intertwined, related, for instance, to publication, promulgation, in force.

To integrate different analyses into operational applications that meet specific needs, trade-offs must be found, perhaps even dynamically, between the depth, re-

7. `http://www.akomantoso.org/`

8. `https://wwwmatthes.in.tum.de/pages/74hcgqw5dmwj/Semantic-Analysis-of-Legal-Texts-SERIT`.

liability, coverage of analysis, and volume of text to process. Unfortunately, integration is considered as a technical rather than as a scientific issue. Various options have been proposed but the constraints are seldom explicit, so the trade offs are rarely documented. Thus it is not known if the proposed solutions are optimal.

Another integration issue is related to the management of ambiguities and errors. The various levels of document and linguistic analysis are interdependent: on the one hand, the ambiguities that appear at one level can be solved at another level, but on the other hand, errors can also be propagated from one level to another, thus impacting the quality of the overall analysis. Therefore, the interactions between the various levels of analysis must be controlled.

Various architectures and workflows have been proposed to tackle some of the NLP integration issues [9]. While, they have not been designed for legal source processing *per se* but legal source analysis is an interesting playground to test and compare their relative strengths and weaknesses, due to its complexity and broad analysis spectrum.

However, integration issues go well beyond text analysis. Applications involve domain knowledge regarding the entities at stake (ontologies) and the reasoning rules. In real applications, NLP is only one of the technologies to be implemented together with semantic technologies, logic and knowledge engineering, data and decision sciences. Developing applications for the various actors involved in legal businesses (legal professionals and their clients, citizens, etc.) is a huge project ahead of us.

5. Presentation of this *TAL* issue

The present issue has two papers that presents two different aspects of research related to legal NLP.

The paper by Jaromir Savelka, Vern R. Walker, Matthias Grabmair and Kevin D. Ashley, entitled "Sentence Boundary Detection in Adjudicatory Decisions in the United States" addresses the specific problem of segmenting legal texts into sentences. It proposes an in-depth analysis of a specific and supposedly simple NLP task, sentence boundary detection, that is a prelude to many more complex ones. This paper illustrates the complexity of legal language and its impact on the quality of the analyses that can be done and on the applications that rely on them.

The second paper focuses on an application. It shows how beneficial a question/answering system on maritime regulations can be for the commander of a boat who needs to know all the regulations that are relevant to a specific type of ship, at a particular time, and in a given space, or for the supervisory authorities in charge of identifying infringements and risk situations on the part of boat commanders. The paper entitled "Un système de question/réponse automatique dans le domaine légal :

9. See, for instance, the General Architecture for Text Engineering (GATE, `http://gate.ac.uk/`), the Natural Language Toolkit (NLTK, `http://www.nltk.org/`) or the Apache Unstructured Information Management Architecture (UIMA, `http://uima.apache.org/`).

le cas des réglementations maritimes", by Yannis Haralambous and Cheikh Kacfah Emani, also shows that NLP, knowledge engineering and semantic web technologies must be combined to develop such an question/answering system.

6. References

Athan T., Governatori G., Palmirani M., Paschke A., Wyner A. Z., "LegalRuleML: Design Principles and Foundations", *in* W. Faber, A. Paschke (eds), *Reasoning Web*, Springer, p. 151-188, 2015.

Bonin F., Dell'Orletta F., Venturi G., Montemagni S., "Singling out Legal Knowledge from World Knowledge. An NLP-based approach", *Proceedings of LOAIT 2010.*, Fietole, Italy, 2010.

Boulet R., Mazzega P., Bourcier D., "A Network Approach to the French System of Legal codes- Part I: Analysis of a Dense Network", *Journal of Artificial Intelligence and Law*, vol. 19, p. 333-355, 2011.

Casanovas P., Palmirani M., Peroni S., van Engers T. M., Vitali F., "Semantic Web for the Legal Domain: The next step", *Semantic Web*, vol. 7, nº 3, p. 213-227, 2016.

Christensen M. L., Olsen H. P., Tarissan F., "Identification of Case Content with Quantitative Network Analysis: An Example from the ECtHR", *in* F. Bex, S. Villata (eds), *Legal Knowledge and Information Systems – JURIX 2016: The 29th Annual Conference*, vol. 294 of *Frontiers in Artificial Intelligence and Applications*, IOS Press, p. 53-62, 2016.

Coulthard M., Johnson A. (eds), *The Routledge handbook of forensic linguistics*, Routledge, 2010.

Dragoni M., Villata S., Rizzi W., Governatori G., "Combining NLP Approaches for Rule Extraction from Legal Documents", *Proceedings of the Workshop on 'MIning and REasoning with Legal texts' collocated at the 29th International Conference on Legal Knowledge and Information Systems*, 2016.

Francesconi E., "Semantic Model for Legal Resources: Annotation and Reasoning over Normative Provisions", *Semantic Web Journal*, vol. 7, nº 3, p. 255-265, 2016.

Gordon T. F., "A Theory Construction Approach To Legal Document Assembly", *In Pre-Proceedings of the 3rd International Conference on Logic, Informatics, and Law*, p. 485-498, 1989.

Höfler S., "Legislative Drafting Guidelines: How Different Are They from Controlled Language Rules for Technical Writing?", *in* T. Kuhn, N. E. Fuchs (eds), *Proceedings of Controlled Natural Language CNL2012*, Springer, p. 138-151, 2012.

Mimouni N., Nazarenko A., Salotti S., "An Approach for Searching and Browsing a Network of Legal Documents", *in* S. F. Radbud Winkels, Nicola Lettieri (ed.), *Network Analysis in Law*, Edizioni Scientifiche Italiane, p. 183-207, 2014.

Moens M.-F., Boiy E., Palau R. M., Reed C., "Automatic Detection of Arguments in Legal Texts", *Proceedings of the 11th International Conference on Artificial Intelligence and Law*, ICAIL '07, ACM, New York, NY, USA, p. 225-230, 2007.

Osslon J., "Forensic linguistics", *in* V. Muhvic-Dimanovski, L. Socanac (eds), *LINGUISTICS – Volume I*, Encyclopedia of life support systems, EOLSS Publishers/UNESCO, p. 378-393, 2009.

Sartor G., Casanovas P., Biasiotti M., Fernndez-Barrera M., *Approaches to Legal Ontologies: Theories, Domains, Methodologies*, Springer Publishing Company, Incorporated, 2013.

Sprowl J., "Automated Assembly of Legal Documents", *in* B. Nesbitt (ed.), *Computer Science and Law – an advanced course*, Cambridge University Press, p. 195-203, 1980.

Winkels R., de Ruyter J., "Survival of the Fittest: Network Analysis of Dutch Supreme Court Cases", *in* M. Palmirani, U. Pagallo, P. Casanovas, G. Sartor (eds), *AI Approaches to the Complexity of Legal Systems (AICOL-III)*, Springer, p. 106-115, Aug., 2011.

Wyner A., Mochales-Palau R., Moens M.-F., Milward D., "Approaches to Text Mining Arguments from Legal Cases", *in* E. Francesconi, S. Montemagni, W. Peters, D. Tiscornia (eds), *Semantic Processing of Legal Texts*, Springer-Verlag, Berlin, Heidelberg, p. 60-79, 2010.

Wyner A., Peters W., "On Rule Extraction from Regulations", *in* K. Atkinson (ed.), *Legal Knowledge and Information Systems – JURIX 2011: The 24th Annual Conference*, IOS Press, p. 113-122, 2011.

Thanks to the members of the specific reading committee of this special issue:

– Kevin Ashley (University of Pittsburgh School of Law, USA)

– Danièle Bourcier (CERSA, CNRS & Université Panthéon-Sorbonne, France)

– Pompeu Casanova (UAB Law School, Universitat Autònoma de Barcelona, Spain)

– Jean-Pierre Chevallet (LIG, Université Grenoble Alpes & CNRS, France)

– Catherine Faron Zucker (I3S, Université de Nice & CNRS, France)

– Aldo Gangemi (LIPN, Université Paris 13 – Sorbonne Paris Cité & CNRS, France)

– Meritxell Fernández-Barrera (ELDA/ELRA, France)

– Thomas Lebarbé (Litt&Arts, Université Grenoble Alpes & CNRS, France)

– François Lévy (LIPN, Université Paris 13 – Sorbonne Paris Cité & CNRS, France)

– Aurélien Max (LIMSI, Université Paris Sud & CNRS, France)

– Marie-Francine Moens (Department of Computer Science, Katholieke Universiteit Leuven, Belgium)

– Isabelle Moulinier (Capital One, Sr. Director, Data Science, USA)

– Monica Palmirani (Computer Science and Law, Bologna University, Italy)

– Wim Peters (Sheffield Natural Language Processing group, UK)

– Patrick Saint-Dizier (IRIT, CNRS & Université Paul Sabatier, France)

– Serena Villata (I3S, CNRS & Nice University, France)

– Giulia Venturi (Institute of Computational Linguistics, CNR, Italy)

– Rabdoub Winkels (Computer Science and Law, University of Amsterdam, Netherlands)

and to the members of the editorial board of *TAL Journal* (see (http://www.atala.org/content/comité-de-rédaction-0)

Sentence Boundary Detection in Adjudicatory Decisions in the United States

Jaromir Savelka* — **Vern R. Walker**** — **Matthias Grabmair***** — **Kevin D. Ashley***

* *University of Pittsburgh*
** *Hofstra University*
*** *Carnegie Mellon University*

ABSTRACT. *We report results of an effort to enable computers to segment US adjudicatory decisions into sentences. We created a data set of 80 court decisions from four different domains. We show that legal decisions are more challenging for existing sentence boundary detection systems than for non-legal texts. Existing sentence boundary detection systems are based on a number of assumptions that do not hold for legal texts, hence their performance is impaired. We show that a general statistical sequence labeling model is capable of learning the definition more efficiently. We have trained a number of conditional random fields models that outperform the traditional sentence boundary detection systems when applied to adjudicatory decisions.*

RÉSUMÉ. *Nous présentons les résultats d'un effort visant à permettre aux ordinateurs de segmenter les décisions arbitrales des États-Unis en phrases. Nous avons créé un ensemble de données de 80 décisions de justice de quatre domaines différents. Nous montrons que les décisions juridiques sont plus difficiles pour les systèmes de détection des limites de peines existantes que pour les textes non juridiques. Les systèmes existants de détection des limites de phrases sont basés sur un certain nombre d'hypothèses qui ne sont pas valables pour les textes légaux, leur performance en est donc altérée. Nous montrons qu'un modèle général d'étiquetage de séquence statistique est capable d'apprendre la définition plus efficacement. Nous avons formé un certain nombre de modèles de champs aléatoires conditionnels qui surpassent les systèmes traditionnels de détection des limites de la peine lorsqu'ils sont appliqués aux décisions juridictionnelles.*

KEYWORDS: *Artificial intelligence and Law, text annotation, sentence boundary detection, conditional random fields, adjudicatory decisions.*

MOTS-CLÉS : *Intelligence artificielle et loi, annotation de texte, détection de limites de phrases, champs aléatoires conditionnels, décisions juridictionnelles.*

1. Introduction

This paper reports results of an effort to enable computers to learn to extract a particular kind of information from legal texts that human readers take for granted: segmenting the texts into sentences that express complete thoughts.

Adjudicatory decisions from the US legal system pose challenges to standard NLP techniques for sentence boundary detection (SBD). Decision makers frequently employ long sentences, complex sentence structures, quotations, citations, and extensive use of parentheses. Citations and lists introduce ambiguities in the meaning of punctuation by using periods and colons that complicate the decision of whether a sentence has ended or not. Researchers have noted that lists, with their use of colons and periods in enumerations and of citations, and their combinations of punctuation and alpha-numeric characters, make it harder to tokenize (regulatory) texts and split them into sentences (Wyner and Peters, 2011). De Maat and Winkels (2009) observed that lists degraded the performance of their sentence classifier.

SBD is a critical task in many applications such as machine translation, summarization, or information retrieval. Presumably, problems in automatically segmenting legal texts into sentences have implications for applying text processing pipelines. Errors in SBD can propagate through higher-level text processing tasks, lowering overall performance. SBD errors are particularly problematic for semantic processing of legal texts that focuses on identifying the inferential roles that sentences play, such as stating legal rules, findings of fact, or a court's conclusion of law. Suboptimal SBD will likely negatively affect the ultimate applications.

We have developed a detailed protocol setting forth guidelines for annotating sentence boundaries in legal decisions. We annotated a data set of 80 court decisions from four domains: cyber-crime decisions, intellectual property decisions, the Board of Veterans' Appeals disability decisions, and decisions of the United States Supreme Court. The complete data set contains more than 26,000 annotations (Section 4) and it is publicly available. [1]

We use the data sets to confirm that legal decisions are more challenging for existing SBD systems than the texts to which they are typically applied (i.e., news articles, short essays). We show that, if the systems are allowed to account for the peculiarities of legal decisions, their performance improves (i.e., if the systems are trained on our data sets). Most importantly, we explain that legal decisions require far more complicated definitions as to what constitutes a sentence compared to other textual data typically used in SBD work. Existing SBD systems are based on a number of assumptions that do not hold for legal text, hence their performance is impaired. We show that a general statistical sequence labeling model, such as conditional random fields (CRF), is capable of learning the definition more efficiently and can significantly outperform the traditional SBD systems in adjudicatory decisions.

1. https://github.com/jsavelka/sbd_adjudicatory_dec

2. Sentence Boundary Detection

The goal in sentence boundary detection is to split a natural language text into individual sentences (i.e., identify each sentence's boundaries). We begin with a standard definition of "sentence" from linguistic theory dealing with written linguistic structures. A sentence is a span of characters consisting of one or more words that are grammatically linked, and which is capable of expressing (at least implicitly) a complete thought. Sentences might express a declarative statement, a question, an exclamation, a request, a command, or a suggestion. A declarative sentence, for example, is an autonomous information unit that is capable of being true or false in a given situation or circumstance (Chierchia and McConnell-Ginet, 2001).

A sentence that explicitly expresses its complete thought typically contains a grammatical subject and a grammatical predicate. The grammatical subject is typically a noun phrase (a group of words that are in dependency relations with a single noun), and refers to the person, place, or thing (including abstract things) that the sentence is about. The grammatical predicate is typically a verb phrase (a group of words that are in dependency relations with a single verb, which in turn refers to an action, process or state). The predicate completes the information about the subject. An example of a normal sentence structure is "The veteran filed a claim for disability benefits," where "the veteran" is the grammatical subject and "filed a claim for disability benefits" is the grammatical predicate.

Not all sentences are as explicit in expressing their complete thoughts. An example of a one-word sentence with implicit meaning is "Yes" when it is an answer to the interrogative sentence "Did you seek medical attention for your condition?" In context, the sentence "Yes" has the same meaning as the explicit sentence "I did seek medical attention for my condition." In Section 3, we discuss other implicit sentence structures (e.g., headings and data fields).

In English, the boundary character that starts a sentence is typically an initial capital letter in the first word of the sentence (i.e., the first character within the span of characters constituting a sentence is a capital letter). Punctuation at the end of the sentence is typically the end character of the sentence span. Sentences in English typically end with one of three punctuation characters: a period (also called a "full stop"), a question mark, and an exclamation mark. In the case where a sentence is enclosed in quotation marks (either single or double), then the quotation marks are included within the sentence boundaries. Similarly, if parentheses (or other brackets) enclose a sentence, then the parentheses are included within the sentence boundaries. However, no annotation span for a sentence should start or end with a white space.

Typically, SBD is operationalized as a binary classification of a fixed number of candidate boundary points (e.g., ".", "!", "?"). For more details see Read *et al.* (2012). Approaches to SBD roughly fall into three categories:

1) *Rules* – A battery of hand-crafted matching rules is applied. The rules may look like the following:

IF "!" OR "?" MATCHED → MARK AS BOUND
(every time there is a "!" or "?" the system should consider it a boundary)
IF "<EOL><EOL>" MATCHED → MARK AS BOUND
(a boundary should be predicted every time the system encounters two consecutive line breaks)

2) *Supervised Machine Learning (ML)* – For each triggering event, decide if it is an instance of sentence boundary. Each event is represented in terms of selected features such as the following:
$x_i =<$ 0:token=".", 0:isTrigger=1, -1:token="Mr",-1:isAbbr=1, 1:token="Lange" $>$
Given the labels $y_i \in \{0, 1\}$ the supervised classifier is a function $f(x_i) \rightarrow y_i$.

Consider the following example:

> In this case, there is no question that the information Mr. Lange offered for sale was a trade secret.

The period after "Mr" is a triggering event but a system can learn that if a period follows an abbreviation then a sentence boundary should be predicted only if the following word starts with a capital letter. Unfortunately this would not work in our example. In addition the system would have to learn that "Mr" is almost always followed by a period and that it almost never ends a sentence. Therefore, the period would not be predicted as a sentence ending token.

3) *Unsupervised ML* – Similar to supervised ML approach but the system is trained on unlabeled data. The system can, for example, recognize that "Mr" is always followed by a period and therefore it is probably an abbreviation which most of the time does not end a sentence.

Multiple SBD systems were reported as having an excellent performance (Read *et al.*, 2012):

– 99.8% accuracy of a decision tree-based classifier in predicting "." as ending (or not) a sentence evaluated on the Brown corpus (Riley, 1989)

– 99.5% accuracy of a combination of an original system based on neural networks and decision trees with and existing system (Aberdeen *et al.*, 1995) evaluated on the *Wall Street Journal* corpus (WSJ) (Palmer and Hearst, 1997)

– 99.75% (WSJ) and 99.64% (Brown) accuracy of a maximum entropy model in assessing ".", "!", and "?" (Reynar and Ratnaparkhi, 1997)

– 99.69% (WSJ) and 99.8% (Brown) accuracy of a rule-based sentence splitter combined with a supervised POS-tagger (Mikheev, 2002)

– 98.35% (WSJ) and 98.98% (Brown) accuracy of an unsupervised system based on identification of abbreviations (Kiss and Strunk, 2006)

Read *et al.* (2012) conducted a study of SBD systems performance across different corpora and report more modest results ranging from 95.0% to 97.6% for different systems. Also, they tested the systems on corpora of user-generated web content. The performance of the SBD systems deteriorated for these corpora where the accuracy often falls in the lower nineties. (Read *et al.*, 2012)

3. Detecting Sentence Boundaries in US Adjudicatory Decisions

Adjudicatory decisions, whether issued by a court or by an administrative tribunal, determine whether a party's conduct conforms to the applicable legal norms, and can impose sanctions or order other remedies when the party has violated those norms. Written adjudicatory decisions are more challenging for SBD than news articles—the traditional subject of interest in developing SBD systems. Whereas news articles are generally short texts, a decision may be short but it may also be as long as a book. A decision may be structured into sections and subsections preceded by a heading (possibly numbered). A decision may contain specific constituents such as a header and a footer, footnotes, or lists. Sentences are interleaved with citations. The sentences themselves may be extremely long, or even partially spread across lists. In decisions there is a high usage of sentence organizers such as ";", or "—" and multiple types of brackets. Quotes are frequent and possibly nested.

Consider the following passage from a decision, which contains one long and complex sentence followed by a citation sentence in parentheses:

> As used in the statute, "'act in furtherance of a person's right of petition or free speech under the United States or California Constitution in connection with a public issue' includes: (1) any written or oral statement or writing made before a legislative, executive, or judicial proceeding, or any other official proceeding authorized by law; (2) any written or oral statement or writing made in connection with an issue under consideration or review by a legislative, executive, or judicial body, or any other official proceeding authorized by law; (3) any written or oral statement or writing made in a place open to the public or a public forum in connection with an issue of public interest; (4) or any other conduct in furtherance of the exercise of the constitutional right of petition or the constitutional right of free speech in connection with a public issue or an issue of public interest." (§425.16, subd. (e), italics added; see Briggs v. Eden Council for Hope & Opportunity (1999) 19 Cal. 4th 1106, 1117-1118, 1123 [81 Cal.Rptr.2d 471, 969 P.2d 564] [discussing types of statements covered by anti-SLAPP statute].)

The first sentence contains a quotation (which in turn contains a second, nested quotation) organized as a list, and it is followed by a sentence of citations and their captions. This text is very challenging for an SBD system because it spans across many triggering events that are not sentence boundaries. The second sentence is a citation and illustrates the occurrence of periods that are not sentence-ending (a common occurrence in adjudicatory decisions). The period character's common use as a sentence boundary can cause extensive segmentation errors in such decisions because it is used copiously for other purposes (e.g., abbreviations or citations).

In annotating adjudicatory texts for sentence boundaries, therefore, it is important to ensure that the annotations provide reliable and valid data. In order to ensure this, the authors adapted and used the protocol for sentence annotation developed by

the Research Laboratory for Law, Logic and Technology (LLT Lab) at the Maurice A. Deane School of Law at Hofstra University.[2] Such annotation protocols provide methods and criteria for manually annotating texts, and a set of conventions governing the generation of annotation data. Protocols are developed in two stages. First, from a sample of documents containing a variety of decisions, examples are collected that display normal forms of the annotation type, linguistic variants of those normal forms, and aberrant forms. Second, those examples are used to derive general guidelines, criteria and conventions for manually annotating these types within texts. Protocols are used not only for manually producing semantic data, but also for assuring the quality of the coding, for replicating experimental results, and for building separate but compatible datasets.

For the annotation type "Sentence", the normal form is a grammatical subject consisting of a noun phrase followed immediately by a grammatical predicate consisting of a verb phrase - i.e., <grammatical subject noun phrase><grammatical predicate verb phrase>. The noun phrase and verb phrase can contain subordinate clauses, provided the sentence as a whole is relatively easy to parse by parts of speech. In general, a span of characters is a "normal form" of an annotation type if we are highly confident that it constitutes an annotation of the specified type, and this confidence is based on some evidence or feature within the span of characters itself (an adequate "linguistic cue"). Also, a sentence in normal form has a certain fixed format or pattern, which we find recurring numerous times. Sentences having a normal form should be the easiest types of sentences for computer software to identify through standard parsing. Examples of sentences in normal form are:

> *The Veteran's chronic adjustment disorder with depressed and anxious features is related to service.*
>
> *The Veteran does meet the criteria for a diagnosis of posttraumatic stress disorder (PTSD).*
>
> *A disability which is aggravated by a service-connected disability may be service-connected.*

A span of text that is a "linguistic transform" of a normal form is one for which we are also confident that it constitutes an annotation of the type "Sentence". This confidence is based on some linguistic cue or feature within the span of text itself. However, while a sentence in normal form has a straightforward format or pattern of <grammatical subject noun phrase><grammatical predicate verb phrase>, a linguistic transform has a linguistic structure that is in principle transformable into one or more sentences that do have normal forms. There might be some linguistic rules that would make it easier for computer software to identify such forms. Examples of linguistic transforms are:

> *Consequently, as outlined in a February 2013 Formal Finding, the RO requested information from the Joint Services Records Research Center*

2. `https://github.com/jsavelka/sbd_adjudicatory_dec`

(JSRRC) and the US Army Crime Records Center; however, those sources provided negative responses for the requested date range.

Establishment of service connection for PTSD in particular requires: (1) medical evidence diagnosing PTSD; (2) credible supporting evidence that the claimed in-service stressor actually occurred; and (3) medical evidence of a link between current symptomatology and the claimed in-service stressor.

See, e.g., Young v. McDonald, 766 F.3d 1348, 1353 (Fed. Cir. 2014) ("PTSD is not the type of medical condition that lay evidence . . . is competent and sufficient to identify.").

Finally, there are spans of text that have a very particular linguistic structure (being neither normal form nor linguistic transform), but we are still confident that they constitute an instance of the type "Sentence". This confidence might be based more on the context than on linguistic cues within the span itself (e.g., co-references with words or phrases in other sentences, or standard conventions within a type of document). The following paragraphs discuss certain classes of examples on which we can generalize for adjudicatory decisions. For each class, because of its distinctive sentence structure, it would be a rather straightforward to develop more semantic information after sentence segmentation.

Case names in document titles express a single thought (one named party is suing another named party), and are best treated as a single sentence. This is true regardless of how the case name is formatted in a particular document (e.g., spread over several lines). For example, the following is a single sentence:

SOUNDEXCHANGE, INC. Plaintiff v. MUZAK, LLC Defendant.

Headings are spans of text that we annotate as "sentences" because they provide information about the organization of the text; they chunk the document into meaningful segments. For example, we understand the heading "FINDINGS OF FACT" as having a meaning similar to "The sentences in the following section state the findings of fact of the tribunal." Other standard headings in adjudicatory documents include:

INTRODUCTION

REASONS AND BASES FOR FINDINGS AND CONCLUSIONS

ORDER

Data fields are spans of text that we annotate as "sentences" because they provide the name of a data field and a value for that field in the particular document or case. They implicitly assert the value of the data field. We understand a data field such as "Decision Date: 03/28/17" as having the same meaning as "This decision was issued on March 28, 2017." Other examples are:

Citation Nr: 1710389

DOCKET NO. 12-12 279

Veteran represented by: Veterans of Foreign Wars of the United States

Page numbers of a reporter service that prints the official version of the adjudicatory decision can occur within the text file in one of two ways. First, if they occur outside of normal sentences, then we annotate them as separate sentences. For example, in the passage below, the character sequence "*1163" means "This is where page 1163 begins in the Federal Reporter, Third Series." This passage therefore contains 3 sentences, the middle one being the sentence "*1163":

> *He contends that to do so would be, in effect, to report himself for the new crime of being found in the country after deportation. *1163 See United States v. Pina-Jaime, 332 F.3d 609, 612 (9th Cir.2003) (holding that an alien need not have reentered the United States illegally to be convicted of being "found in" the country illegally).*

Second, a page number could occur embedded within a normal sentence, wherever the page break happens to fall in the printed version. In such a situation, we do not split the normal sentence into parts just because a page number happens to occur inside it. We can deal with the embedded page number in subsequent analyses, after sentence segmentation. For example, the following is segmented as a single sentence containing the page number "*1162":

> *The record in this case shows no attempt, by either the Probation Officer or sentencing court, to justify this *1162 sweeping condition.*

Ellipses (...) also occur in one of two ways. First, if the ellipsis occurs within a sentence span and it indicates missing words from within that sentence, then the ellipsis should be included within the overall sentence span. For example, the following is a single sentence that begins a block quotation, and the ellipsis occurs within the sentence boundaries:

> *... the conferee's objective was to limit the grandfather to their existing services in the same transmission medium and to any new services in a new transmission medium where only transmissions similar to their existing service are provided.*

Second, if the ellipsis occurs between sentences, then such an ellipsis should be annotated as a separate sentence. The rationale is that the ellipsis provides coded information ("Sentences have been deleted"), and should not be parsed within the complete sentences that precede or follow it. For example, the following passage contains two ellipses that we annotate as separate sentences (the first ellipsis occurs after a completed sentence and the other one occurs between paragraphs in the block quote):

> *3. The defendant shall comply with the immigration rules and regulations of the United States, and, if deported from this country, either voluntarily or involuntarily, not reenter the United States illegally. The defendant is not required to report to the Probation Office while residing out-side *1157 of the United States; however, within 72 hours of release from any custody or any reentry to the United States during the period of Court-tordered supervision, the defendant shall report for instructions to the*

United States Probation Office. . . .
. . .
5. The defendant shall not access or possess any computer or computer-related devices in any manner, or for any purpose, unless approved in advance by the Probation Officer.

Phrases functioning as complete sentences are annotated as complete sentences, as though there is an implicit ellipsis. For example, the following is only a noun phrase, but because it occurs in a list with the heading "ISSUES", we understand it to have the same meaning as the sentence "An issue in this case is whether there is entitlement to service connection for a psychiatric disorder.":

Entitlement to service connection for a psychiatric disorder.

Parentheticals within sentences occur frequently within adjudicatory decisions in the United States, especially within citations. We annotate the parenthetical as within the span of the overall sentence. This is the treatment even if, as occasionally happens, the parenthetical itself contains one or more separate sentences (i.e., the sentences within the parentheses are not annotated separately). For example, the following is a single sentence:

Id. at 576, 128 S. Ct. 558; see also id. at 575, 128 S. Ct. 558 ("The District Court began by properly calculating and considering the advisory Guidelines range. It then addressed the relevant § 3553(a) factors.").[6]

Colons as sentence-ending punctuation can sometimes occur as an exception to the normal presumption that a colon is not sentence-ending punctuation. This one exceptional situation is when the colon is the last punctuation mark in a paragraph block of text—i.e., the colon is followed immediately by a line break. A colon is therefore treated as sentence-ending punctuation if, but only if, it is followed immediately by a line break.

There are several reasons for making this exception. Although the use of a colon can be highly stylistic, in general an author uses a colon instead of a period to express that what goes before the colon is meaningfully related to what comes after – that they are in effect connected into one thought. That is why the colon is presumptively not sentence-ending punctuation. However, an author may use a colon followed immediately by a line break to introduce a block quote or an enumerated list of items. In such a situation, if we do not end the sentence with the colon, there may be no good place to end it. A block quote might contain multiple sentences or paragraphs. To include the entire block quote within the boundaries of the sentence that happens to introduce the quote would leave the block quote unsegmented. Moreover, the introductory sentence should be parsed separately from the block quote, and may have no meaning in common with the quote itself. The quoted sentence (or sentences) should be annotated independently, and parsed separately, without being part of the sentence that introduces the block quote. Similarly, a stand-alone enumerated list introduced by a colon followed by a line break should be annotated independently of the introducing sentence.

Unfortunately, a colon followed immediately by a line break might separate a grammatical subject from a list of grammatical verbs, as we sometimes find in quotations from statutory or regulatory texts. The convention we adopt here is a compromise between the desire to have first-pass segmentation that is as non-semantic as possible (comparable to tokenization) and the desire to preserve intact all propositional content in the process of sentence segmentation. We stress that this is a first-pass compromise. After this initial segmentation into sentences, on subsequent passes we can parse the spans of text before and after a <colon><line-break> to determine if they are semantically related, and if warranted we can then annotate the entire passage as (also) a single, overall sentence.

The following are two examples in which the span of the introductory sentence ends with the colon, because the colon is followed immediately by a line break:

> *For example, in a June 1977 service personnel record a counselor opined that:*
>
> *I have personally interviewed this SM and found him to have a good attitude towards the Army. However, he has a serious academic problem.*
>
> *Accordingly, the case is REMANDED for the following action:*
>
> *1. When disability ratings and effective dates have been determined for all service-connected disabilities, to include those granted herein, and all development that the RO deems necessary is undertaken, the Veteran's request for a TDIU should be readjudicated.*

Enumerated lists (whether numbered or lettered) require special treatment, and the treatment depends on whether the list items are themselves sentences or not.

If the list items are themselves sentences (including headings that we annotate as sentences), then we annotate the list number or letter itself (i.e., the number or letter of the list item) as itself a sentence, and the sentence that is the list item as another sentence. The rationale is that we would create problems for machine learning if we include the list number (e.g., "1.") as part of the sentence that is the list item. An ML program should not treat "1." as part of the sentence it introduces, or try to POS parse the sentence including the "1." within the sentence boundaries. Moreover, the "1." expresses a thought separate from the sentence it introduces. The numbering of the list could change, but the role and meaning of each sentence on the list would remain the same. For example, the following passage consists of five sentences (the heading, two list numbers, and two sentences that are list items):

> *FINDINGS OF FACT*
> *1. The Veteran does meet the criteria for a diagnosis of posttraumatic stress disorder (PTSD).*
> *2. The Veteran's chronic adjustment disorder with depressed and anxious features is related to service.*

If the list items are not themselves sentences, then there is one overall sentence that includes the list items, and the list numbers or letters occur within that overall sentence. In such a case, there is only one sentence. For example, the following is a single sentence containing an enumerated list:

> *Establishment of service connection for PTSD in particular requires: (1) medical evidence diagnosing PTSD; (2) credible supporting evidence that the claimed in-service stressor actually occurred; and (3) medical evidence of a link between current symptomatology and the claimed in-service stressor.*

Although some sentences quoted from statutes and regulations are very long and complex, we follow this same instruction in annotating them (e.g., when we annotate a block quotation of a statute or regulation that occurs within an adjudicatory decision). For example, the following consists of two sentences (the first sentence ends with the colon followed immediately by a line break):

> *These factors are:*
> *(1) the nature and circumstances of the offense and the history and characteristics of the defendant;*
> *(2) the need for the sentence imposed*
> *(A) to reflect the seriousness of the offense, to promote respect for the law, and to provide just punishment for the offense;*
> *(B) to afford adequate deterrence to criminal conduct;*
> *(C) to protect the public from further crimes of the defendant; and*
> *(D) to provide the defendant with needed educational or vocational training, medical care, or other correctional treatment in the most effective manner;*
> *(3) the kinds of sentences available;*
> *(4) the kinds of sentence and the sentencing range established for . . . the applicable category of offense committed by the applicable category of defendant as set forth in the guidelines . . .*
> *(5) any pertinent policy statement . . . issued by the Sentencing Commission . . . subject to any amendments made to such policy statement by act of Congress. . . .*
> *(6) the need to avoid unwarranted sentence disparities among defendants with similar records who have been found guilty of similar conduct; and*
> *(7) the need to provide restitution to any victims of the offense.*

Endnotes or footnotes present annotation challenges in two ways. First, the in-text indicators for endnotes or footnotes (usually numbers, but sometimes letters or other characters) should be included within the boundaries of the sentence where they occur. Sometimes they are embedded within the span of the sentence, and sometimes they occur after the sentence-ending punctuation. In the latter situation, they are still annotated as being within the span of the sentence. For example, each of the following is a single sentence (the number in square brackets being the endnote indicators):

> *Barsumyan was arrested and indicted for one count of producing, using, and trafficking in a counterfeit credit card, 18 U.S.C. § 1029(a)(1),*

> *and three counts of possession of device-making equipment, 18 U.S.C. § 1029(a)(4).[2]*
>
> *Barsumyan gave the Agent a "skimming device," [1] and asked her to covertly "skim" the hotel guests' credit cards when they registered.*

Second, if the endnotes themselves appear as a numbered list (e.g., at the end of the decision), then the annotation follows the instructions for numbered lists. The following passage would consist of four sentences (with "[4]" being the first sentence, meaning "The fourth endnote is the following.", followed by two normal sentences and a citation sentence):

> *[4] Both wireless telephones and credit cards are considered "access devices" for these purposes. The cloning of wireless telephones was considered particularly serious because cloned cell phones are commonly used by drug dealers and other criminals to evade surveillance. See 144 Cong. Rec. S3021 (1998) (statement of Sen. Leahy); 143 Cong. Rec. S2655 (1997) (statement of Sen. Kyl).*

Grammatical or typographical errors sometimes occur. Occasionally we can still determine from context and the content of the span that the span of text is a sentence (e.g., often because it is followed by a heading or a standard sentence), even if it contains grammatical or typographical errors. In such a case, we still annotate it as a sentence. For example, the following should be annotated as a sentence, despite the missing period at the end, because this span of characters was followed by a normal sentence:

> *38 U.S.C.A. § 1111 (West 2014)*

4. Data Set

We assembled a data set consisting of 80 court and administrative decisions. These came from four distinct areas of law (20 decisions from each)—appeals of veterans' disability decisions (BVA), cyber crime (CC), intellectual property (IP), and decisions of the Supreme Court of the United States (SCOTUS). We briefly describe these four data sets as well as the document selection processes in the subsections below. Selected summary statistics are provided in Table 1. The data set is publicly available. [3]

Four human annotators (the authors) marked sentence boundaries in the decisions. We were guided by the annotation protocol described earlier (Section 3). To increase the quality and consistency of the annotations we used the automatic SBD system developed by some of the authors in prior work (Savelka and Ashley, 2017). Each decision was marked by one of the annotators. First, the automatic segmenter was applied. The task of the human annotator was to correct its output.

We have double-annotated 2 randomly selected decisions from each of the areas to measure inter-annotator agreement (i.e., 8 decisions with more than 2,500 sentences).

3. `https://github.com/jsavelka/sbd_adjudicatory_dec`

		# total	longest doc	average doc	shortest doc
BVA	chars	474,478	76,255	23,723.9	9,555
(20 docs)	tokens	170,166	28,493	8,508.3	3,351
	sents	3,727	568	186.4	80
Cyber crime	chars	984,756	181,009	49,237.8	16,859
(20 docs)	tokens	367,740	71,653	18,387.0	5,986
	sents	8,295	1,613	414.8	134
IP	chars	932,133	103,974	46,606.7	15,877
(20 docs)	tokens	343,831	38,536	17,191.6	6,204
	sents	7,262	724	363.1	90
SCOTUS	chars	960,890	85,175	48,044.5	5,621
(20 docs)	tokens	355,677	31,872	17,783.9	2,130
	sents	6,768	602	338.4	62
Total	chars	3,352,257	181,009	41,903.2	5,621
(80 docs)	tokens	1,237,414	71,653	15,467.7	2,130
	sents	26,052	1,613	325.7	62

Table 1. *Summary statistics of the four data sets and their aggregate. Statistics are reported on the level of characters (chars), tokens and sentences (sents).*

The inter-annotator agreement for the different areas of law, as well as the overall agreement, is reported in Table 2. It is important to emphasize the use of the automatic SBD system in the annotation process. The agreement is most likely somewhat higher than it would be if the process was fully manual. In order to produce disagreement, one of the annotators must conclude that the automatic segmenter erred and correct its output while the other one considers it correct. If both of the human annotators correct the output, the disagreement is produced if the corrections differ.

The agreement was evaluated from two different perspectives. First, a match could be declared if:

1) *boundaries* – we count each boundary on its own; or

2) *segments* – both boundaries need to match.

Second, for each of these perspectives, two approaches could be used to determine if the boundary was predicted correctly. A match would be declared only if:

1) *strict* – boundary offsets match exactly; or

2) *lenient* – the difference between boundary offsets does not contain an alphanumeric character.

Let us consider the following example where |T| stands for the true boundary and |P| for a predicted boundary:

> |T||P|Accordingly, we find that the circuit court did not abuse its discretion when it denied Mr.|P| |P|Renfrow's motion for a JNOV.|T| |T|**|P|We find no merit to this issue.|T||P|

	BVA	Cyber Crime	IP	SCOTUS	Overall
strict-sen	.95	.93	.90	.90	.91
lenient-sen	.96	.93	.90	.91	.92
strict-bound	.97	.95	.94	.95	.95
lenient-bound	.98	.96	.95	.95	.95

Table 2. *Inter-annotator agreement.*

Two of the predicted boundaries match the true boundaries. The remaining three differ. In case of one of the three, the difference subsists in the two asterisks (non-alphanumeric). From the strict boundaries perspective (strict-bound), the Precision (P) is 0.4 and Recall (R) is 0.5. Using the lenient-boundaries perspective (lenient-bound), the P is 0.6 and R is 0.75. From the strict-segments perspective (strict-seg), both P and R are 0 (no segment is predicted correctly). Using the lenient-segments perspective (lenient-seg), the P is 0.33 and R is 0.5. Using the different perspectives allows more detailed analysis of the agreement. As shown above, a decent agreement on two boundaries does not necessarily imply that a whole segment is also predicted correctly.

Board of Veterans' Appeals Disability Decisions

We selected a sample of 20 decisions on compensation for service-related disabilities, issued by the Board of Veterans' Appeals (BVA), which is an administrative appellate tribunal within the US Department of Veterans Affairs (VA). The VA administers benefits for veterans of the US Uniformed Services, such as disability compensation, educational assistance, and other benefits. The BVA's workload has increased dramatically in the past few decades, e.g., reaching 55,713 decisions in fiscal year 2015 (Moshiashwili, 2014) (Board of Veterans' Appeals, 2015). The vast majority of appeals considered by the BVA involve claims for disability compensation (Board of Veterans' Appeals, 2015).

The decisions in this dataset are specialized to compensation for service-related disabilities, but the content of the decisions resembles the typical content of trial-level judicial decisions, with descriptions of the procedural history of the case, conclusions of law about the applicable legal rules, citations to authority and to the evidentiary record, extensive review of the evidence in the case, explanation of the tribunal's reasoning, and findings of fact on the critical legal issues. The BVA has the statutory authority to decide the facts of each case *de novo* (Moshiashwili, 2014), and it must provide a written statement of the reasons or bases for its findings and conclusions. That statement "must account for the evidence which [the BVA] finds to be persuasive or unpersuasive, analyze the credibility and probative value of all material evidence submitted by and on behalf of a claimant, and provide the reasons for its rejection of any such evidence." *Caluza v. Brown*, 7 Vet.App. 498, 506 (1995), *aff'd*, 78 F.3d 604 (Fed. Cir. 1996).

Cyber Crime, Intellectual Property, and Supreme Court Decisions

The remaining data sets comprise judicial decisions from different appellate and trial courts, times, and subject matters.

The cyber crime data set comprises of 20 decisions from criminal proceedings where the alleged offense had a strong connection to cyber space or IT technology in general. The typical offenses may involve credit card frauds, possession and distribution of electronic child pornography, or cyber bullying. The decisions were retrieved from freely accessible on-line services such as Court Listener [4] and Google Scholar [5] on the basis of hand-crafted search queries. An example query could look like this: "cybercrime unit". Because the decisions involve criminal proceedings they often emphasize fact finding and evidential reasoning.

The 20 decisions of the US Supreme Court span nearly 200 years, from the 1803 decision in Marbury v. Madison, establish the principle of judicial review, to a 2001 decision concerning a statute of limitations under the Fair Credit Reporting Act. Decisions in between those dates deal with due process and equal protection in segregation cases, the right to boycott, the WW II detention of US citizens of Japanese descent and Congress's war powers, due process and citizenship, search and seizure, the Commerce Clause and civil rights, the right to assistance of counsel under the 6th amendment, freedom of expression, birth control, presidential Executive privilege, the right to privacy, and assisted suicide. The formats of the decisions include "slip opinions," a version the Court publishes shortly after releasing a bench opinion. These may include corrections and deal with some page breaks in a complex way. We have selected some of the landmark decisions listed in the dedicated Wikipedia entry. [6]

The 20 intellectual property and related cases comprise a mix of US Supreme Court, federal Court of Appeals and federal district court cases involving issues under the federal Patent Act, 1976 Copyright Act, the Lanham Act on trademark law, and the Electronic Computer Privacy Act. Part of the data set are recent more prominent IP cases. The rest are older cases related to the IP protection of computer programs.

5. Experimental Design

We conducted a series of experiments to test several hypotheses. The first hypothesis (i.) is that court and administrative decisions are more challenging for SBD than traditional texts such as news articles. We measure performance of existing vanilla SBD systems (i.e., using the pre-trained general models) on BVA, Cyber Crime, IP, and SCOTUS data sets. The hypothesis is tested by means of comparing the measured performance with the performance of the systems reported for other types of texts (see Section 2).

4. `www.courtlistener.com`
5. `scholar.google.com`
6. `en.wikipedia.org/wiki/Lists_of_United_States_Supreme_Court_cases`

As explained in Section 3, adjudicatory decisions are subject to a number of linguistic peculiarities that make SBD a challenge. Where possible we train the existing SBD systems to explore how well they adjust to legal decisions. Specifically, we assess the hypothesis (ii.) that, by enabling the systems to account for some of the linguistic peculiarities commonly found in the decisions, their performance improves. We test the hypothesis by comparing the performance of the trained systems with that of the systems using the pre-trained general models.

There are certain assumptions about sentence boundaries that are useful for SBD on general English texts (Section 2). For example, it is useful to understand SBD in terms of binary classification of a finite set of triggering events (e.g., ".", "!", and "?") as to whether they constitute a sentence boundary or not. We test the hypothesis (iii.) that operating under these assumptions in case of decisions hurts the SBD performance by preventing the systems from accounting for some of the phenomena that regularly occur in legal texts. We implement an SBD system consisting of a general condition random fields sequence labeling model (CRF; for details see Section 6). The system does not start with any assumptions as to what could constitute a sentence. It learns the rules exclusively by means of training on labeled data. We test the hypothesis by comparing the performance of this system to the performance of the existing systems trained on our data sets.

We explore if there are peculiarities specific to different areas of law (hypothesis iv.). The point is to find out if it is more important to use decisions from the same or closely related area of law, or if it is feaible to train one general SBD system on decisions from different areas. Specifically, we train the traditional SBD models as well as the custom CRF model on one of the data sets and apply them to other data sets. We test the hypothesis by comparing the performance of the systems on the documents from the same data set to the performance on the documents from the other data sets.

For evaluation we use traditional information retrieval metrics—precision (P), recall (R), and F_1-measure (F_1). The comparison is done at the micro level, meaning that statistics are computed over all sentences across all documents in a given collection. The measures based on the alternate match criteria tend to correlate quite well. Therefore, for the sake of clarity, we only report the lenient-boundary-focused approach (one of the approaches described in Section 4).

6. Results

Vanilla SBD Systems

For evaluation of SBD systems' performance on the corpora of adjudicatory decisions we use one system from each category:

1) As an example of a system based on rules, we worked with the SBD module from the Stanford CoreNLP toolkit (Manning *et al.*, 2014). [7]

2) To test a system based on supervised ML classifier, we employed the SBD component from openNLP. [8]

3) As an example of an unsupervised system, we used the punkt (Kiss and Strunk, 2006) module from the NLTK toolkit. [9]

The criterion for selection of the SBD systems was that they are components of, we assume, widely used general NLP toolkits.

The *rule-based sentence splitter* from Stanford CoreNLP requires a text to be already segmented into tokens. The system is based on triggering events, the presence of which is a prerequisite for a boundary to be predicted. The default events are a single "." or a sequence of "?" and "!". The system may use information about paragraph boundaries which can be configured as either a single EOL (i.e., line break) or two consecutive EOLs. The system may also exploit HTML or XML markup if present. Certain patterns that may appear after a boundary are treated as parts of the preceding sentence (e.g., parenthesized expression).

The *supervised sentence splitter* from OpenNLP is based on a maximum entropy model which requires a corpus annotated with sentence boundaries. The triggering events are ".", "?", and "!". As features the system uses information about the token containing the potential boundary and about its immediate neighbors:

– the prefix

– the suffix

– the presence of particular chars in the prefix and suffix

– whether the candidate is an honorific or corporate designator

– features of the words left and right of the candidate. (Ratnaparkhi, 1998)

The *unsupervised sentence splitter* (punkt) from NLTK does not depend on any additional resources besides the corpus it is supposed to segment into sentences. The leading idea behind the system is that the chief source of wrongly predicted boundaries are periods after abbreviations. The system discovers abbreviations by testing the hypothesis $P(\cdot|w) = 0.99$ against the corpus. Additionally, token length (abbreviations are short) and the presence of internal periods are taken into account. For prediction the system uses:

– orthographic features

– a collocation heuristic (collocation is evidence against split)

– a frequent sentence starter heuristic (split after abbreviation). (Kiss and Strunk, 2006)

7. `nlp.stanford.edu/software/corenlp.shtml`
8. `opennlp.apache.org`
9. `nltk.org/api/nltk.tokenize.html`

	BVA			CC			IP			SCOTUS			Overall		
	P	R	F_1	P	R	F_1	P	R	F_1	P	R	F_1	P	R	F_1
CoreNLP	.77	.84	.81	.80	.76	.78	.77	.81	.79	.77	.76	.76	.78	.78	.78
punkt	.68	.84	.75	.72	.79	.75	.69	.80	.74	.69	.80	.74	.70	.80	.75
openNLP	.77	.81	.79	.79	.75	.77	.80	.80	.80	.77	.78	.78	.78	.78	.78

Table 3. *Vanilla SBD systems performance.*

The results of application of the three SBD systems on the four SBD data sets described in Section 4 are summarized in Table 3. The results clearly show that performance of the general SBD systems is drastically lower when compared to the performance on news articles data sets. It is also much below the reported performance on the user-generated web content. (Section 2 or Read *et al.*, 2012, for more details.) Certain portions of this gap could be explained by the particular definition of the SBD task we adopt. The remaining portion is due to the decisions being particularly challenging for SBD.

Trained SBD Systems

OpenNLP and punkt may be trained on a custom data set, which is encouraged. It can be expected that such training will improve performance of these two systems. We use the data from each data set with labeled sentence boundaries to train data-set-specialized openNLP and punkt models (^{BVA}openNLP+, ^{IP}punkt+, etc.). We also use a pooled data set (all four data set combined) to train generalized models (^{G}openNLP+ and ^{G}punkt+).

It should be noted that punkt is an unsupervised system and as such it does not use the labels in its training. Therefore, training punkt is very cheap and one could use a training set of much greater size. Indeed, we expect that if we use a larger data set to train punkt, its performance would increase beyond what we observe in our experiments. The same does not hold for openNLP, which is trained in a supervised fashion from the gold labels. Training openNLP is quite expensive. If we would wish to use more documents in training (increasing the performance further), we would have to manually label additional documents.

The CoreNLP SBD module is rule-based and therefore it is not possible to train a custom model. To approximate training, one could use its configuration options and tune the system to perform well on our data set. We configured the CoreNLP SBD module to perform well on the data sets and evaluated it alongside trained openNLP and punkt models. It should be emphasized that this kind of comparison is very problematic and it should be taken with a grain of salt. Specifically, the authors were familiar with the documents. In light of this familiarity, it appears that the CoreNLP could have an unfair advantage in this experiment.

	BVA			CC			IP			SCOTUS			Overall		
	P	R	F_1	P	R	F_1	P	R	F_1	P	R	F_1	P	R	F_1
^{BVA}punkt+	.94	.84	.89	.73	.74	.74	.68	.80	.73	.65	.80	.72	.72	.79	.75
^{CC}punkt+	.79	.86	.82	.82	.74	.78	.76	.79	.78	.71	.79	.75	.77	.79	.78
^{IP}punkt+	.80	.86	.83	.80	.74	.77	.80	.79	.80	.72	.79	.75	.78	.78	.78
^{SC}punkt+	.76	.86	.80	.79	.74	.77	.74	.80	.77	.77	.78	.78	.77	.79	.78
^{G}punkt+	.94	.85	.89	.80	.75	.77	.79	.80	.79	.75	.79	.77	.80	.79	.82
^{BVA}openNLP+	.95	.84	.89	.83	.70	.76	.83	.75	.79	.83	.73	.78	.85	.74	.79
^{CC}openNLP+	.87	.83	.85	.91	.76	.83	.90	.81	.85	.88	.78	.83	.89	.79	.84
^{IP}openNLP+	.96	.83	.89	.88	.74	.80	.93	.82	.87	.91	.77	.83	.91	.78	.84
^{SC}openNLP+	.93	.80	.86	.85	.72	.78	.88	.77	.82	.92	.79	.85	.89	.76	.82
^{G}openNLP+	.96	.84	.90	.92	.76	.83	.93	.82	.87	.92	.79	.85	.93	.80	.86
CoreNLP+	.79	.96	.87	.84	.90	.87	.80	.91	.85	.78	.83	.81	.81	.90	.85

Table 4. *Trained SBD systems performance.*

The performance of the trained (or configured) systems is summarized in Table 4. We observe that all the systems perform better when compared to the vanilla versions. The performance of some of the systems on some of the data sets is in the mid-eighties. This is still much lower than the performance reported for news articles and the performance of the models on user-generated web content (Read *et al.*, 2012).

Custom SBD Systems

We created two simple custom SBD systems. One is based on a set of handcrafted rules while the other one uses machine learning to infer the rules from our data sets. The rule-based system first replaces all sentence ending punctuation with a masking character if it occurs in an environment matching at least one of a list of manually defined regular expressions of typical legal document punctuation patterns. In a second step, the document is traversed beginning to end and begin-end-pairs are gathered for each detected sentence. It should be noted that the extracted sentences are not technically guaranteed to cover the full document. This segmenter and its set of masking regular expressions was created during a different project focusing on US Trade Secret Law decisions which had no overlapping documents with the experiments reported in this paper.

As the second system, we trained a number of conditional random fields models based on simple low-level textual features. In prior work, we showed that more complex features help to improve the performance of the system even further (Savelka and Ashley, 2017). We do not deal with this issue here and we reserve fine-tuning of the models for future work. A CRF is a random field model that is globally conditioned on an observation sequence O. The states of the model correspond to event labels E. We use a first-order CRF in our experiments (observation O_i is associated with E_i).

We use the CRFsuite [10] implementation of first-order CRF (Lafferty *et al.*, 2001; Liu *et al.*, 2005; Okazaki, 2007).

We use a very aggressive tokenization strategy that segments text into a greater number of tokens than usual. The reason for this is to capture tokens such as a single or double line breaks that may be very suggestive about sentence ending. We consider an individual token to be any consecutive sequence consisting entirely of one type of character, using the following character types:

1) letters

2) numbers

3) whitespace.

Each character that does not belong to any of the above constitutes a single token. For example, the following sequence is tokenized as shown below:

Call me at 9am on my phone (123)456-7890.

[“Call”, “ ”, “me”, “ ”, “at”, “9”, “am”, “ ”, “on”, “ ”, “my”, “phone”, “ ”, “(”, “123”, “)”, “456”, “-”, “7890”, “.”]

Each of the tokens is then a data point in a sequence that a CRF model operates on.

Each token is represented by a small set of relatively simple features. Specifically, the set includes:

1) *lower* – a token in lower case.

2) *sig* – a feature representing a signature of a token. This feature corresponds to the token with the following transformations applied:

a) each lower case letter is rewritten to “c”

b) each upper case letter is rewritten to “C”

c) each digit is rewritten to “D”.

3) *length* – a number corresponding to the length of the token in characters if the length is smaller than 4. If the length is between 4 and 6 the feature is set to “normal.” If it is greater than 6 it is set to “long.”

4) *islower* – a binary feature which is set to true if all the token characters are in lower case.

5) *isupper* – a binary feature which is set to true if all the token characters are in upper case.

6) *istitle* – a binary feature which is set to true if the first of the token characters is in upper case and the rest in lower case.

7) *isdigit* – a binary feature which is set to true if all the token characters are digits.

8) *isspace* – a binary feature which is set to true if all the token characters are whitespace.

10. `www.chokkan.org/software/crfsuite/`

	BVA			CC			IP			SCOTUS			Overall		
	P	R	F_1	P	R	F_1	P	R	F_1	P	R	F_1	P	R	F_1
Custom Rules	.91	.90	.90	.74	.76	.75	.78	.82	.80	.75	.75	.75	.78	.79	.79
^{BVA}CRF	.99	.98	.99	.87	.63	.73	.87	.66	.75	.86	.65	.74	.89	.74	.77
^{CC}CRF	.96	.90	.93	.96	.92	.94	.96	.95	.96	.97	.96	.96	.95	.95	.95
^{IP}CRF	.97	.90	.93	.96	.90	.93	.97	.95	.96	.97	.94	.95	.96	.94	.95
^{SC}CRF	.95	.87	.91	.94	.90	.92	.93	.93	.93	.97	.94	.95	.95	.93	.94
^{G}CRF	.99	.99	.99	.95	.94	.95	.95	.96	.95	.97	.96	.96	.97	.95	.96

Table 5. *Custom SBD systems performance.*

In addition, for each token we also include *lower*, *sig*, *islower*, *isupper*, *istitle*, *isdigit*, and *isspace* features from the three preceding tokens and three following tokens. If one of these tokens falls beyond the document boundaries, we signal this by including *BOS* (beginning of sequence) and *EOS* (end of sequence) features.

Taking a look at the "Call me at 9am ..." sequence from the above example, the third token of this sequence ("me") would be represented along the following lines:

```
{bias, 0:lower=me, 0:sig=cc, 0:length=2, 0:islower=true,
 0:isupper=false, 0:istitle=false, 0:isdigit=false,
 0:isspace=false, -3:BOS, -2:lower=call, -2:sig=Cccc,
 -2:length=normal, -2:islower=false, -2:isupper=false,
 -2:istitle=true, -2:isdigit=false, -2:isspace=false,
 -1:lower=" ", -1:sig=" ", -1:length=1, -1:islower=false,
 -1:isupper=false, -1:istitle=false, -1:isdigit=false,
 -1:isspace=true, 1:lower=" ", 1:sig=" ", 1:length=1,
 1:islower=false, 1:isupper=false, 1:istitle=false,
 1:isdigit=false, 1:isspace=true, 2:lower=at, 2:sig=cc,
 2:length=2, 2:islower=true, 2:isupper=false, 2:istitle=false,
 2:isdigit=false, 2:isspace=false, 3:lower=" ", 3:sig=" ",
 3:length=1, 3:islower=false, 3:isupper=false, 3:istitle=false,
 3:isdigit=false, 3:isspace=true}
```

As labels we use the Sentence annotation type projected into the BILOU[11] as demonstrated on the following example:

> Look! It is here.
>
> ["Look", "!, " ", "It", " ", "is", " ", "here", "."]
>
> [B-Sentence, L-Sentence, O, B-Sentence, I-Sentence, I-Sentence, I-Sentence, I-Sentence, L-Sentence]

11. B: beginning of sequence, I: inside sequence, L: last in sequence, O: outside of sequence, U: unit-length sequence.

The performance of the custom SBD systems is reported in Table 5. For most of the data sets the performance of some of the models reaches the middle nineties. This performance is comparable to the performance of the traditional SBD models on user-generated web content (Read *et al.*, 2012). For a small number of data sets, the performance is in the higher nineties comparable to the performance of SBD systems on news articles (Read *et al.*, 2012).

7. Discussion

The results of applying off-the-shelf SBD systems on legal decisions (Table 3) clearly show that the performance of the general SBD systems is drastically lower as compared to their performance on news articles data sets. It is also much below the reported performance on the user generated web content (Read *et al.*, 2012). Certain portions of this gap could be explained by the particular definition of the SBD task we adopt (Section 3). The remaining portion is due to the decisions being particularly challenging for SBD.

The most common source of errors is due to wrongly predicted sentence boundaries in citations as shown in the example:

> see United States v. X-Citement Video, Inc., 513 U.S. 64, 76-78, 115 S. Ct. 464, 130 L. Ed.|P| 2d 372 (1994)

The predicted boundary is marked with |P|. This type of error is very serious because it causes broken sentences to be passed along for further processing within the pipeline. These sentences may eventually even show up in the output presented to a user (e.g., in a summary).

Another commonly occurring type of error is a missed boundary that follows a unit if a triggering event is absent:

> 1)|T| Response to Jury Question|T|

The true boundary is marked with |T|; the absence of a predicted boundary |P| indicates an error. This type of error is partly caused by our specific definition of SBD. This type of mistake is less serious than the previous one. It may still negatively affect the performance of the processing pipeline but it does not introduce broken sentences that may eventually be output to a user.

The performance of the trained (or configured) systems universally improved over their off-the-shelf counterparts (compare Table 4 and Table 3). Even though the performance of CoreNLP improved, the wrongly predicted boundaries in citations remain a problem. Below are two examples of the boundaries that were incorrectly predicted by CoreNLP+:

1) Entick v. Carrington, 95 Eng.|P| Rep. 807 (C. P. 1765)

2) 451 F. Supp.|P| 2d 71, 88 (2006).

The training improved the performance of the general openNLP SBD module dramatically when it comes to precision. The performance in terms of recall remained about the same. Although some of the boundaries are missed, it is quite rare for openNLP+ to predict an incorrect boundary. Systematic errors are mostly missed boundaries such as those in the following examples:

1) 5. The Government's Hybrid Theory|T|

2) This device delivers many different types of communication: live conversations, voice mail, pages, text messages, e-mail, alarms, internet, video, photos, dialing, signaling, etc.|T| The legal standard for government access depends entirely upon the type of communication involved.

In the first example the system missed a boundary because it is not associated with a triggering event (heading). Example 2 is interesting because the system obviously learned that the "etc." is an abbreviation which often does not end a sentence.

The trained punkt+ performs better than the general one. It still commits slightly more errors as compared to the other two trained/configured systems. One would probably need to train punkt+ on a considerably larger data set in order to match the performance of the other two systems. The previously identified typical errors occur:

1) II. ANALYSIS|T|

2) "[T]he district court retains broad discretion in deciding how to respond to a question propounded from the jury and . . . |P| the court has an obligation to dispel any confusion quickly and with concrete accuracy."

Example 1 shows a missed boundary after a heading. Example 2 shows a wrongly predicted boundary after three dots in a quotation.

The custom CRF system clearly outperforms both vanilla and trained general systems on all four data sets, suggesting that our general hypothesis holds. Although the system performs quite well, there is certainly room for improvement. Here are some examples of errors in predicting boundaries:

1) Such a procedure, this Court said, "cannot be an adequate substitute for the right to full appellate review available to all defendants"|P| *743 who may not be able to afford such an expense.

2) *654|T| III.

3) "The introduction of this article declares the opinion.|P| . . that Congress could not declare"

4) "It settles the great question of citizenship and removes all doubt as to what persons are or are not citizens of the United States.|T| . . . We desired to put this question of citizenship and the rights of citizens .|P| . . under the civil rights bill beyond the legislative power . . . ."

Both examples 1 and 2 relate to the phenomenon of editorial content inserted into the text of a decision. These are page numbers indicating that there is a page break in

a printed document. Dealing with this phenomenon is difficult because we treat these as standalone sentences if they fall in between two other sentences but we ignore them if they are embedded within a single sentence. Examples 3 and 4 show mishandling of ellipses. Dealing with ellipses is difficult and they are a source of many errors.

8. Future Work

For future work we would like to use the data set that we have assembled to train more powerful sentence boundary detectors. Despite the nice improvement over the traditional SBD systems, the number of errors is still considerable. The prediction model that we used here is quite simple (a single CRF model). We have already shown that chaining multiple models together improves the SBD performance (Savelka and Ashley, 2017). There the focus was on distinguishing the main and the auxiliary content first and then using this information in decisions about sentence boundaries. A similar setup could probably lead to even better results than those reported in Savelka and Ashley (2017) because we now have a significantly richer and larger data set. In addition, more recent (and presumably more effective) sequence labeling models than CRFs could be employed (e.g., long short-term memory networks).

9. Conclusion

We assembled a data set consisting of 80 court and administrative decisions. These came from four distinct areas of law. We annotated the decisions with sentence boundaries producing a data set that consists of more than 24,000 sentences. We used the data set to show that court and administrative decisions are more challenging for SBD than traditional texts such as news articles. This is due to some peculiar linguistic features that regularly occur in adjudicatory decisions. We confirmed this by training the available SBD systems on our data set observing a visible improvement in performance. We also explored the usefulness of typical assumptions traditional SBD systems operate on. We found that operating under these assumptions for legal decisions hurts the SBD performance. It prevents the systems from accounting for some of the phenomena that regularly occur in legal texts. A general CRF model trained on our data set performed significantly better than the traditional SBD systems.

Acknowledgements

This work was supported in part by the National Institute of Justice Graduate Student Fellowship (Fellow: Jaromir Savelka) Award # 2016-R2-CX-0010, "Recommendation System for Statutory Interpretation in Cybercrime." This work was also supported in part by the Maurice A. Deane School of Law at Hofstra University, through its general support of the Research Laboratory for Law, Logic and Technology and through its support of this particular research project.

10. References

Aberdeen J., Burger J., Day D., Hirschman L., Robinson P., Vilain M., "MITRE: description of the Alembic system used for MUC-6", *Proceedings of the 6th Conference on Message Understanding*, Association for Computational Linguistics, p. 141-155, 1995.

Board of Veterans' Appeals U. S. D. o. V. A., "Annual Report: Fiscal Year 2015", 2015.

Chierchia G., McConnell-Ginet S., *Meaning and grammar: An introduction to semantics*, Second Edition, MIT press, 2001.

de Maat E., Winkels R., "A next step towards automated modelling of sources of law", *Proceedings of the 12th International Conference on AI and Law*, ACM, p. 31-39, 2009.

Kiss T., Strunk J., "Unsupervised multilingual sentence boundary detection", *Computational Linguistics*, vol. 32, n° 4, p. 485-525, 2006.

Lafferty J., McCallum A., Pereira F. *et al.*, "Conditional random fields: Probabilistic models for segmenting and labeling sequence data", *Proceedings of the 18th International Conference on Machine Learning, ICML*, vol. 1, p. 282-289, 2001.

Liu Y., Stolcke A., Shriberg E., Harper M., "Using conditional random fields for sentence boundary detection in speech", *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, Association for Computational Linguistics, p. 451-458, 2005.

Manning C. D., Surdeanu M., Bauer J., Finkel J. R., Bethard S., McClosky D., "The stanford corenlp natural language processing toolkit", *ACL*, p. 55-60, 2014.

Mikheev A., "Periods, capitalized words, etc.", *Computational Linguistics*, vol. 28, n° 3, p. 289-318, 2002.

Moshiashwili V. H., "The Downfall of Auer Deference: Veterans Law at the Federal Circuit in 2014", *Am. UL Rev.*, vol. 64, p. 1007, 2014.

Okazaki N., "CRFsuite: a fast implementation of Conditional Random Fields", 2007.

Palmer D. D., Hearst M. A., "Adaptive multilingual sentence boundary disambiguation", *Computational Linguistics*, vol. 23, n° 2, p. 241-267, 1997.

Ratnaparkhi A., Maximum entropy models for natural language ambiguity resolution, PhD thesis, University of Pennsylvania, 1998.

Read J., Dridan R., Oepen S., Solberg L. J., "Sentence boundary detection: A long solved problem?", *COLING (Posters)*, vol. 12, p. 985-994, 2012.

Reynar J. C., Ratnaparkhi A., "A maximum entropy approach to identifying sentence boundaries", *Proceedings of the 5th Conference on Applied Natural Language Processing*, Association for Computational Linguistics, p. 16-19, 1997.

Riley M. D., "Some applications of tree-based modelling to speech and language", *Proceedings of the Workshop on Speech and Natural Language*, ACL, p. 339-352, 1989.

Savelka J., Ashley K. D., "Using Conditional Random Fields to Detect Different Functional Types of Content in Decisions of United States Courts with Example Application to Sentence Boundary Detection", *Proceedings of the 2nd Workshop on Automated Detection, Extraction and Analysis of Semantic Information in Legal Texts*, ICAIL, 2017.

Wyner A., Peters W., "On Rule Extraction from Regulations.", *JURIX*, vol. 11, p. 113-122, 2011.

Un système de questions-réponses automatiques dans le domaine légal : le cas des réglementations maritimes

Cheikh KACFAH EMANI* — **Yannis HARALAMBOUS****

* cheikh.kacfah@imt-atlantique.fr, *IMT Atlantique, CS 83818, 29238 Brest Cedex 3 et DECIDE, UMR CNRS 6285 Lab-STICC.*

** yannis.haralambous@imt-atlantique.fr, *IMT Atlantique, CS 83818, 29238 Brest Cedex 3 et DECIDE, UMR CNRS 6285 Lab-STICC.*

RÉSUMÉ. *Nous présentons les premiers travaux du projet REIZHMOR dont le but est la modélisation de textes juridiques et réglementaires autour de la navigation maritime. Après une présentation du corpus, constitué par les arrêtés préfectoraux et interpréfectoraux référencés dans un volume des* Instructions nautiques *du Shom (Service hydrographique et océanographique de la marine), nous décrivons l'élaboration d'un système de questions-réponses fondé sur des requêtes SPARQL adressées à une base de connaissances, avec une attention particulière portée aux difficultés spécifiques du langage juridique.*

ABSTRACT. *We present the first steps of the REIZHMOR project, the goal of which is to model legal and regulatory texts on sea navigation. After a short presentation of the corpus, consisting of prefectoral decrees referenced in a volume of the* Sailing Directions *of the Hydrographic and Oceanographic Service of French Navy, we describe the development of a Question-Answering system based on SPARQL queries to a knowledge base, giving particular attention to the specific difficulties of legal language.*

MOTS-CLÉS : *textes réglementaires, textes juridiques, navigation maritime, ontologie légale, système questions-réponses, SPARQL, règles.*

KEYWORDS: *Regulatory texts, legal texts, sea navigation, legal ontology, Question-Answering system, SPARQL, rules.*

1. Introduction

De nombreux textes régissent le domaine de la navigation maritime (de Cet Bertin, 2008). En ce qui concerne la navigation dans les eaux françaises et dans les eaux internationales, ces textes proviennent de divers corpus tels que les règles et recommandations internationales applicables aux transports et activités maritimes rédigées par l'Organisation maritime internationale (MARPOL, SOLAS, etc.), le code des ports maritimes, les arrêtés des préfectures maritimes, etc. En guise d'aide à la navigation, à ce corpus s'ajoutent les *Instructions nautiques* du Shom (Service hydrographique et océanographique de la marine) ainsi que les *Avis aux navigateurs*. Les *Instructions nautiques* sont des textes accompagnant les cartes marines ; elles complètent ces dernières en fournissant des informations telles que la réglementation, les données météorologiques, les canaux à très hautes fréquences ou même des informations culturelles et linguistiques (Sauvage-Vincent, 2017 ; Haralambous *et al.*, 2017). Les *Avis aux navigateurs* sont des informations de sécurité maritime qui mettent à jour ou complètent les documents nautiques, de façon permanente ou temporaire. Nous appellerons ce corpus, les *réglementations maritimes*.

Dans cet ensemble complexe de réglementations, il est primordial pour le navigateur – qui est aux commandes d'un type de navire précis, à un moment donné et dans un espace donné – de connaître l'ensemble des réglementations qui sont pertinentes pour lui. D'un autre côté, il est important pour les autorités chargées de la surveillance des espaces maritimes et des équipements afférents à ceux-ci, de relever les infractions et les situations à risque de la part des navigateurs. Enfin, les réglementations évoluant sans cesse, il est important pour les rédacteurs juridiques de s'assurer que l'ensemble des réglementations ne présente pas de contradiction et que l'ajout de nouveaux textes préserve sa cohérence.

Le Shom est en train de mettre en place une base de connaissances (Sauvage-Vincent, 2017 ; Haralambous *et al.*, 2017) à partir des informations contenues dans les *Instructions nautiques* et les autres ouvrages qu'il publie. Il est prévu que cette base de connaissances soit accessible aux navigateurs et leur propose un certain nombre de services contextuels, c'est-à-dire tenant compte de la nature et de la situation du navire, de sa position et des conditions météorologiques. Le but du projet REIZHMOR (mot-valise signifiant « loi maritime » en breton), démarré en avril 2017 et financé par le Shom, est d'ajouter un module juridique et réglementaire à cette base de connaissances, de manière à exploiter également le corpus réglementaire maritime.

Dans cet article, nous nous intéressons à la tâche qui consiste à répondre automatiquement aux questions qui peuvent être posées en langage naturel à la base de connaissances. Ce procédé est appelé *réponses automatiques aux questions* (RAQ), (*Question Answering* en anglais). Notre but est de poser les bases d'un système pour mener à bien cette tâche.

En tant que preuve de concept, nous avons sélectionné et traité un corpus (§ 2), nous nous sommes appuyés sur des travaux existants concernant une ontologie maritime en les adaptant à nos besoins (§ 3.1), et nous avons élaboré un système de

Liberté • Égalité • Fraternité
RÉPUBLIQUE FRANÇAISE

PRÉFECTURE MARITIME DE LA MANCHE ET DE LA MER DU NORD

Cherbourg, le 23 février 2017

PRÉFECTURE MARITIME DE LA MANCHE ET DE LA MER DU NORD
Division « action de l'État en mer »

ARRÊTÉ PRÉFECTORAL N° 03/2017

RÉGLEMENTANT LE DÉPÔT D'ENGINS SUSPECTS TROUVÉS EN MER ET FIXANT LES ZONES DE DÉPÔT TEMPORAIRE ET DE NEUTRALISATION DE CES ENGINS AUX ABORDS DES PRINCIPAUX PORTS DE LA FAÇADE MANCHE - MER DU NORD

-

Le vice-amiral d'escadre Pascal Ausseur
préfet maritime de la Manche et de la mer du Nord,

Vu la convention des Nations Unies sur le droit de la mer du 10 décembre 1982 ;

Vu le code des transports ;

Vu le code pénal ;

Vu l'ordonnance n° 2012-1218 du 2 novembre 2012 portant réforme pénale en matière maritime ;

Vu le décret n° 76-225 du 4 mars 1976 fixant les attributions respectives du ministre de l'Intérieur et du ministre de la Défense en matière de recherche, de neutralisation, d'enlèvement et de destruction des munitions et explosifs ;

Vu le décret n° 2004-112 du 6 février 2004 modifié, relatif à l'organisation de l'action de l'État en mer ;

Considérant que tout engin trouvé en mer ayant les caractéristiques ou l'aspect d'un matériel susceptible de contenir des matières explosives doit être considéré comme dangereux.

ARRÊTE

Article 1er : Définition des engins suspects.

Sont considérés comme engins suspects tous les objets trouvés en mer, immergés ou flottants, susceptibles de contenir des matières explosives dangereuses, notamment ceux ayant les caractéristiques ou l'aspect d'un matériel de guerre (bombe, obus, mine, torpille).

Préfecture maritime de la Manche et de la mer du Nord – CC 01 – 50115 Cherbourg-en-Cotentin Cedex
Tél : 02.33.92.60.61 – Fax : 02.33.92.59.26 – pac.aem@premar-manche.gouv.fr

Figure 1. *Exemple de la première page d'un arrêté*

réponses automatiques aux questions fondé sur des patrons et débouchant sur des requêtes SPARQL (§ 6).

Dans la section suivante, nous décrivons notre corpus.

2. Le corpus textuel : les arrêtés référencés dans l'*Instruction nautique* C2A

Pour rester aussi près que possible de la base de connaissances du Shom dont la source principale sont les *Instructions nautiques*, nous avons décidé de sélectionner en tant que corpus un ensemble d'arrêtés préfectoraux et interpréfectoraux référencés et en partie reproduits dans un même volume des dites *Instructions*. Nous avons choisi le volume C2A (Shom, 2010), qui traite les côtes nord et ouest de la France, entre la frontière belge et la pointe de Penmarc'h. Nous avons ainsi récupéré et traité 75 arrêtés préfectoraux et interpréfectoraux (préfectures maritimes de Brest et de Cherbourg) qui s'étalent chronologiquement entre le 10 juin 1963 et le 23 février 2017.

2.1. *Structure d'un arrêté préfectoral*

Les arrêtés de notre corpus ont tous la même structure logique et visuelle (fig. 1) qui peut être résumée comme suit : (1) un intitulé (le plus souvent une phrase utilisant un verbe au participe présent : « Arrêté interdisant la navigation à proximité de... », cf. § 2.1.1) ; (2) un ou plusieurs signataires (indiquant la fonction du signataire, et souvent aussi son identité : « Le vice-amiral d'escadre Pascal Ausseur, préfet maritime de la Manche et de la mer du Nord ») ; (3) les *visas* : il s'agit de références vers d'autres textes commençant invariablement par le mot « VU » (souvent écrit en majuscules et/ou en gras), cf. § 2.1.2 ; (4) les « SUR DEMANDE » ou « SUR PROPOSITION » : il s'agit de personnes ou d'institutions ayant formulé une demande ou une proposition qui est à l'origine de l'arrêté ; (5) les « CONSIDÉRANT » : à la suite des visas, ils indiquent les motivations de l'arrêté (par exemple : « CONSIDÉRANT que tout engin trouvé en mer [...] doit être considéré comme dangereux. ») ; (6) le mot « ARRÊTE » (ou « ARRÊTENT » dans le cas de plusieurs signataires), invariablement écrit en majuscules ; (7) les *articles* de l'arrêté, dont l'avant-dernier concerne souvent la gestion des infractions à l'arrêté et le dernier les personnes ou institutions chargées de son exécution ; (8) des éventuelles *annexes*. Dans la majorité des cas, les annexes précisent, par des listes de coordonnées géographiques ou par des cartes, la zone d'application de l'arrêté.

2.1.1. *Les intitulés*

Les intitulés des arrêtés de notre corpus sont tous, à deux exceptions près, des phrases utilisant des verbes au participe présent : « réglementant » (ou « portant règlement ») dans 45 % des cas ; « interdisant » (ou « portant interdiction ») dans 19 % des cas, l'objet de l'interdiction pouvant être la navigation, le mouillage, le dragage, le chalutage, le rejet à la mer d'objets, la plongée sous-marine, la pêche, la baignade, ou les activités aquatiques et subaquatiques ; « portant création d'une zone » dans 11 % des cas, la zone pouvant être interdite, réglementée, d'immersion de déblais de dragage ou de mouillage ; « portant définition d'une zone » dans 7 % des cas ; « instituant », « autorisant », « délimitant », « précisant » et « portant restriction » dans quelques cas isolés.

Notons que dans 7 % de cas l'intitulé décrit l'arrêté comme étant « relatif » à des sujets très spécifiques (à la circulation des navires, au compte-rendu obligatoire des navires, à l'accès aux ports, au pilotage des bateaux et à la navigation des bateaux fluviaux).

2.1.2. *Les visas*

Il s'agit de références intertextuelles, énumérées dans un ordre qui respecte *grosso modo* la hiérarchie des normes de droit français : règlements internationaux et conven-

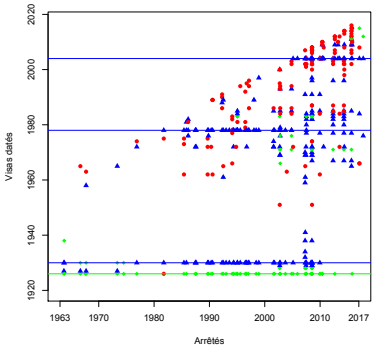


Figure 2. *Correspondance entre arrêtés et dates de visas (losanges verts : lois, triangles bleus : décrets, disques rouges : arrêtés)*

tions[1], codes, ordonnances, lois et décrets, arrêtés, demandes administratives, procès-verbaux et avis de personnes.

Dans la figure 2 nous avons représenté pour chaque arrêté (les arrêtés étant disposés chronologiquement sur l'abscisse) les dates des visas de type loi, décret et arrêté qu'il comporte. On constate que depuis les années 2000 la proportion d'arrêtés figurant dans les visas augmente, et les dates de ceux-ci sont assez proches de la date de publication de l'arrêté. La distribution des décrets reste assez uniforme, alors que les visas de lois sont plutôt épars et se concentrent dans la période après 2000 pour les arrêtés, et les années 70 pour les lois. Nous avons remarqué une présence récurrente de quatre visas pour lesquels nous avons tracé sur la figure des lignes horizontales indicatrices :

(1) la loi du 17 décembre 1926 portant code disciplinaire et pénal de la marine marchande (60 % des arrêtés du corpus) ; (2) le décret du 1er février 1930, relatif aux pouvoirs de police et à la réglementation de la pêche côtière (76 % des arrêtés du corpus) ; (3) le décret nº 78-272 du 9 mars 1978 relatif à l'organisation des actions de l'État en mer (59 % des arrêtés qui lui sont postérieurs) ; (4) le décret nº 2004-112 du 06 février 2004 relatif à l'organisation de l'action de l'État en mer (48 % des arrêtés qui lui sont postérieurs). À cela s'ajoute l'ordonnance royale du 14 juin 1844 concernant le service de la marine (police des rades) qui est quasiment omniprésente (83 % des arrêtés du corpus) mais que nous n'avons pas représentée sur le diagramme pour rendre la partie 1920-2017 plus lisible.

Il est intéressant de noter que même si la structure des arrêtés est restée la même, on remarque de fortes variations dans leur volume, ainsi qu'une certaine évolution au fil du temps.

1. Dans les arrêtés des années 60 nous avons observé des références indirectes vers les conventions internationales à travers de décrets les publiant. Ce n'est qu'à partir de 2002 que nous observons des références directes vers de telles conventions.

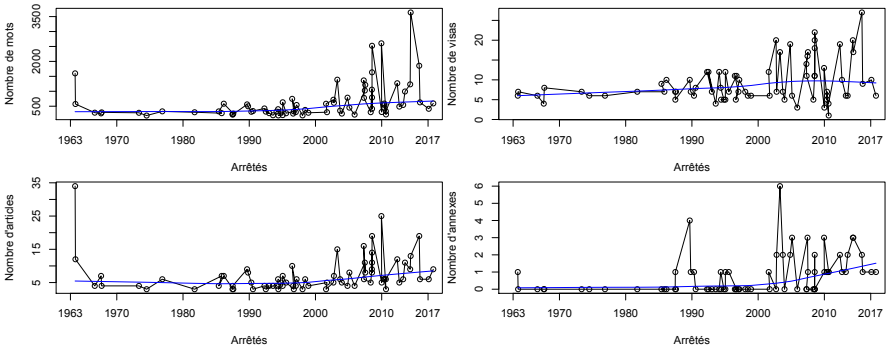


Figure 3. *Évolution du nombre de mots, de visas, d'articles et d'annexes dans le corpus d'arrêtés*

2.1.3. *Évolution de la volumétrie des arrêtés*

La figure 3 représente l'évolution du nombre de mots, de visas, d'articles et d'annexes de notre corpus d'arrêtés. Les courbes bleues représentent des courbes de régression quadratique. À l'exception de l'arrêté du 10 juin 1963 « portant règlement de police et de sécurité pour l'arsenal de Brest et pour les établissements maritimes de l'arrondissement maritime de Brest », qui est exceptionnel par sa taille (1 600 mots, 34 articles) et de l'arrêté du 9 juin 1989 « réglementant l'accès à l'île de Cézembre (Ille-et-Vilaine) » qui comporte quatre annexes, les arrêtés de la période 1960-2000 sont plutôt réduits en nombre de mots, visas, articles, annexes. À partir de l'an 2000, on constate une très grande variabilité des quatre paramètres. Globalement nous constatons une très légère augmentation pour les trois premiers paramètres, et une augmentation un peu plus significative du nombre d'annexes.

3. Modélisation ontologique du corpus

3.1. *L'ontologie du projet* e-Compliance

Un des livrables du projet européen *e-Compliance* (2014) a été une ébauche d'*ontologie maritime* (Lohrmann *et al.*, 2014). Les concepts de cette ontologie sont répartis en quatre catégories (appelées « classes ») : (1) le « légal » : les réglementations, les documents les contenant, le sens qui peut être extrait d'une réglementation. C'est dans cette catégorie que l'on trouve le concept central de l'ontologie qui est `Rule` ; (2) le « maritime » : types de navires et activités maritimes afférentes aux réglementations ; (3) l'« organisationnel » : les organisations, les juridictions et les rôles qui sont responsables des réglementations ou concernés par elles ; (4) le « territorial » : les espaces (géographique, politique, légal) dans lesquels les réglementations s'appliquent et dans lesquels une action prend place.

Dans cette ontologie, une réglementation est un ensemble de *clauses*, et à partir des clauses on peut extraire des *règles* atomiques. Une règle a trois parties : (1) une *cible*, `target`, qui est l'« objet » sur lequel la prescription ou la recommandation est faite ; (2) une *exigence*, `requirement`, qui dénote l'action requise par la cible pour être conforme à la réglementation ; (3) optionnellement, un *contexte*, `context`, précisant les conditions d'application de la règle.

Par exemple, dans la clause « l'accès de toute personne en état d'ivresse est interdit », la cible est « toute personne en état d'ivresse » et l'exigence est l'interdiction d'accès. Notons que la terminologie de cette ontologie reste limitée. Elle nécessite une extension au corpus qui nous intéresse.

3.2. *Extension de l'ontologie* e-Compliance *au corpus des arrêtés*

Après nettoyage du corpus (dans lequel nous n'avons gardé que le contenu des intitulés et des articles des arrêtés) nous avons procédé à une extraction terminologique des termes complexes à l'aide du logiciel *Acabit* (Daille, 2003). Celui-ci nous a fourni 3 650 candidats que nous avons vérifié manuellement pour aboutir à 1 164 termes complexes (du type groupe nominal) d'une longueur moyenne de 2,976 mots (le terme le plus long étant « règlement général de police, de navigation, de mouillage et de pêche »).

Parallèlement à cela nous avons extrait les arbres syntaxiques par dépendances des textes à l'aide du logiciel *MaltParser* (Nivre *et al.*, 2006). Ensuite nous avons intégré les termes complexes dans les arbres syntaxiques en écrasant les sous-arbres correspondant à des termes complexes. Finalement nous avons traduit l'information extraite dans le format OWL, en utilisant des techniques introduites par Simperl *et al.* (2008) pour l'élaboration d'une ontologie juridique. Cela nous a permis d'étendre l'ontologie e-Compliance en tenant compte des connaissances extraites de notre corpus textuel.

4. Les réponses automatiques aux questions et leurs difficultés

Avec Hirschman et Gaizauskas (2001) nous disons qu'un système de RAQ est « un système qui permet à des utilisateurs de poser leurs questions en langage naturel, en utilisant leur propre terminologie, et pour lesquelles ils attendent des réponses précises, obtenues en interrogeant une base de connaissances ». La problématique de la RAQ a été abordée très tôt dans le domaine de l'intelligence artificielle dans les années 70 (Lopez *et al.*, 2011). Néanmoins, malgré l'ancienneté du domaine, plusieurs défis restent encore à relever. Dans le sous-domaine du Web sémantique (Berners-Lee *et al.*, 2001), il existe plusieurs dizaines de travaux abordant la RAQ. Le récent état de l'art proposé par Höffner *et al.* (2016) permet de se rendre compte de la diversité des travaux dans ce domaine[2] et surtout d'exhiber les défis inhérents à tous les systèmes

2. Ainsi, Höffner *et al.* (2016) ont identifié plus de 72 publications proposant 62 systèmes de RAQ. Ces publications couvrent la période de novembre 2010 à juillet 2015.

de RAQ. Ainsi, (Höffner *et al.*, 2016) regroupent les difficultés que doivent surmonter les systèmes de RAQ en sept catégories :

1) la *variété lexicale* ;

2) l'*ambiguïté* induite par la polysémie des termes ;

3) le *multilinguisme* ;

4) la *complexité de la question.* S'il est facile de répondre à une question « simple », dans le sens où elle peut être modélisée par un seul triplet RDF[3] (*e.g.* « quel est l'intitulé de l'arrêté 03/2017 ? »), les questions complexes peuvent, quant à elles, faire appel à plusieurs triplets RDF qu'il faut identifier correctement pour les combiner de manière adéquate et si nécessaire inclure des filtres, des fonctions d'agrégation ou des tris ;

5) les *bases de connaissances distribuées.* Dans certains cas, il est nécessaire de combiner les informations de plusieurs sources pour pouvoir répondre à une question de l'utilisateur. Adresser ce problème peut nécessiter l'exploitation d'alignements déjà existants entre différentes ressources ou alors l'utilisation d'un moteur d'inférence pour les obtenir à la volée ;

6) les *questions procédurales, temporelles ou spatiales.* Les bases de connaissances fondées sur les triplets RDF se prêtent difficilement aux questions temporelles (par exemple concernant l'ordonnancement des événements), aux questions spatiales (par exemple sur le degré de superposition d'entités géographiques) et aux questions procédurales (c'est-à-dire celles qui demandent une liste d'étapes, autrement dit : une procédure) afin de résoudre un problème ;

7) les *patrons de questions.* Pour les questions complexes (voir item 4 ci-dessus), plusieurs approches ont recours aux *patrons* (cf. définition ci-dessous) syntaxiques et/ou sémantiques pour extraire le sens de la question.

Dans un domaine donné, nous nous proposons de décrire l'ensemble des questions valides en tant que langage formel : toute question est alors la séquence de feuilles d'un arbre de dérivation, à condition qu'elles soient des symboles terminaux de la grammaire, et donc des membres de l'alphabet du langage. Nous appelons *patron* d'une question une séquence de feuilles d'un arbre de dérivation, *constituée de symboles non terminaux* (par exemple, pour la phrase « la fille mange la pomme », des patrons possibles sont « GN GV », « GN V GN », etc.). Intuitivement on peut dire qu'un patron est le résultat d'une suite de règles de production appliquées à l'axiome de départ sans qu'on « aille jusqu'au bout », c'est-à-dire sans qu'on n'aboutisse à des symboles terminaux.

Une approche de RAQ privilégiée par les chercheurs est la génération de patrons de requêtes SPARQL à partir de patrons de questions (le formalisme SPARQL étant choisi parce qu'il fait partie intégrante de nombreux outils du Web sémantique).

3. Un triplet RDF est un triplet de ressources « sujet, prédicat, objet » modélisant une phrase du type « sujet, verbe, complément ».

5. Particularités de la RAQ liées au domaine réglementaire

En plus des difficultés mentionnées dans la section précédente, un système de RAQ ciblant une base de connaissances réglementaires doit faire face à des spécificités induites par les particularités du langage juridique (qui peut être considéré comme étant un langage contrôlé, cf. § 5.1) et le décalage d'articulation conceptuelle existant dans les ontologies légales (cf. § 5.2).

5.1. *Le langage réglementaire juridique en tant que langage contrôlé*

Pour le lecteur non spécialiste, le langage juridique, et plus particulièrement, le langage législatif (Cornu, 2005, titre 2, chap. 1) semble rigide (au sens où, du moins dans le cadre de la réglementation, l'utilisation de la périphrase est très limitée et que tout mot ajouté ou supprimé est susceptible d'avoir un impact fort sur la sémantique de la phrase), voire quasi formel et proche du langage des mathématiques. Néanmoins, cette « formalité » apparente ne fait pas du langage juridique un véritable langage formel.

Pour l'évaluer néanmoins en tant que *langage contrôlé*, nous utilisons la classification PENS de Kuhn (2014).

5.1.1. *La précision*

Selon le cas (type de document réglementaire, auteur du document, cadre juridique), on peut dire que du point de vue de la *précision* on se situe entre le P^2 (langages imprécis : *degree of ambiguity and vagueness is considerably lower than in natural languages, and their interpretation depends much less on context*[4]) et le P^3 (langages à interprétation fiable : *syntax is heavily restricted, though not necessarily formally defined. The restrictions are strong enough to make automatic interpretation reliable*[5])[6].

4. Traduction : le degré d'ambiguïté et d'imprécision est considérablement inférieur à celui des langages naturels, et leur interprétation dépend beaucoup moins du contexte.

5. Traduction : la syntaxe est fortement réduite même si elle n'est pas nécessairement formellement définie. Les restrictions sont suffisamment fortes pour rendre l'interprétation automatique fiable.

6. Pour illustrer la variabilité (qui est faible mais néanmoins présente) du langage juridique dans notre corpus, prenons un article que l'on retrouve obligatoirement dans chaque arrêté et qui exprime le fait que les infractions à l'arrêté seront sanctionnées selon un certain nombre de textes indiqués. Le sens de cet article est invariable, pourtant sa formulation peut prendre plusieurs formes : « Les infractions au présent arrêté sont passibles des peines prévues par... », « Les infractions au présent arrêté exposent leurs auteurs aux poursuites et aux peines prévues à... », « Les infractions au présent arrêté [...] feront l'objet de poursuites conformément à... », « Ces infractions sont punies des peines prévues par les mêmes codes », « Les infractions au présent arrêté sont prévues et réprimées par... ». On constate donc que le même sujet (« les infractions ») peut être utilisé avec les verbes « être passible », « exposer ses auteurs », « être puni », « être prévu et réprimé ». La variabilité est surtout lexicale, mais on constate aussi, dans certains des cas ci-dessus, l'emploi d'une métonymie : on dit que « les infractions sont punies »

5.1.2. *L'expressivité*

Les critères qui permettent d'évaluer l'expressivité d'un langage contrôlé sont : la quantification universelle de premier ordre (sur les individus), l'arité des relations, l'existence de structures de règle (si ... alors), l'existence de la négation, la quantification universelle du deuxième ordre (sur les concepts et les relations). Clairement, l'éloquence de ses auteurs et la richesse de la langue française situent le langage juridique en E^5 (langages d'expressivité maximale).

5.1.3. *La naturalité*

Le langage juridique est un cas typique de langage N^4 (*languages with sentences that can be considered valid natural sentences. Speakers of the respective natural language recognize the statements as sentences of their language and are able to correctly understand their essence without instructions or training*[7]) à cela près que les francophones non spécialistes ne possèdent pas forcément le vocabulaire spécifique et ne peuvent donc accéder que partiellement à la sémantique du texte (il s'agit de l'« écran linguistique » dont parle Cornu (2005, p. 12)).

5.1.4. *La simplicité*

Le langage juridique est un langage S^1 (langage très complexe) puisqu'il a la complexité de la langue française *sans restriction syntaxique ou sémantique*[8].

Nous concluons donc que le langage juridique, considéré en tant que langage contrôlé, se situe dans la zone $P^{2\text{-}3}E^5N^4S^1$ de la classification de Kuhn (2014)[9].

5.2. *Le décalage d'articulation ontologique*

Le second défi que doivent relever les systèmes de RAQ et qui est prégnant dans le domaine juridique est la différence entre l'articulation des concepts et des rela-

alors que ce sont, en réalité, leurs auteurs qui le sont. Enfin on constate de la variabilité au niveau de l'utilisation du verbe « prévoir » : dans un cas ce sont les peines qui sont prévues dans les textes, dans un autre cas ce sont les infractions qui le sont. Cela montre les défauts de formalité d'une formule qui pourtant semble figée et est souvent répétée à l'identique, tel un *mantra*, pendant des décennies.

7. Traduction : langages avec des phrases qui peuvent être considérées comme des phrases valides de langage naturel. Les locuteurs du langage naturel correspondant reconnaissent les assertions en tant que phrases de leur langue et sont capables de comprendre leur essence correctement sans instructions ou formation préalable.

8. Citons Cornu (2005, p. 316) : « le langage juridique français ne s'oppose pas à la langue française ; il la met en œuvre ».

9. Notons qu'il existe un langage contrôlé dans le domaine législatif ayant à peu près la même classification Kuhn : $P^2E^5N^5S^1$, il s'agit du *Massachusetts Legislative Drafting Language*, qui est défini par une centaine de règles syntaxiques, sémantiques et structurelles à appliquer à la langue anglaise. Il a été introduit en 2003 par le sénat de Massachusetts (Massachusetts Senate, 2010).

tions dont on trouve des représentations lexicales dans le texte et celle des concepts et relations présents dans l'ontologie. Cette différence est un obstacle à l'alignement direct entre les concepts et relations extraits du texte et ceux prévus dans l'ontologie, nous l'appellerons *décalage d'articulation ontologique*. En guise d'illustration, prenons la règle « Tout pétrolier d'un tonnage supérieur à 150 000 tonnes doit être muni du Certificat de prévention de la pollution des eaux de mer par les hydrocarbures ». Cette exigence fait intervenir : (a) une entité représentée lexicalement par le terme `"pétrolier"`, ayant un attribut pour représenter la notion de `tonnage` et sur lequel on a imposé la condition `tonnage` > 150 ; (b) une entité pour capturer le sens du terme `"Certificat de prévention de la pollution des eaux de mer par les hydrocarbures"` ; (c) une *relation directe* entre les deux entités mentionnées ci-dessus (le premier doit être muni du deuxième).

Or, lorsque l'on considère l'ontologie maritime proposée par Lohrmann *et al.* (2014), décrite en section 3.1, le schéma conceptuel est différent : pour modéliser la règle à travers l'ontologie maritime d'e-Compliance, l'entité dénotant `"pétrolier"` est la cible et l'entité dénotant `"Certificat de prévention de la pollution des eaux de mer par les hydrocarbures"` est l'exigence ; les deux entités étant connectées à une instance de la classe `Rule`.

En utilisant la syntaxe Turtle[10], ces assertions se présentent ainsi[11] :

```
:Ship1 a :Ship;
    :shipType "pétrolier";
    :minTonnage "150".
:Certificate1 a :Certificate;
    :documentTitle "Certificat de prévention de la pollution...".
:Rule1 a :Rule,
    :hasTarget :Ship1;
    :hasRequirement :Certificate1.
```

On voit que `"pétrolier"` n'a pas le statut d'entité mais est une simple valeur d'attribut. De même, on est contraint de passer par une utilisation du concept de règle `Rule` à travers les propriétés `:hasTarget`, `:hasRequirement`, `:documentTitle`, `:shipType`, non présents lexicalement dans la règle textuelle de départ (le statut de règle étant implicite dans la phrase « Tout pétrolier... », un affirmatif que Cornu (2005, p. 274)appelle une *marque ostensible de généralité*).

Cet exemple permet de se rendre compte de fait qu'une approche de RAQ ayant affaire à une base de connaissances construite sur une ontologie légale doit être capable de résoudre le décalage d'articulation ontologique inhérent à celle-ci. Dans la section

10. `https://www.w3.org/TR/turtle/`.

11. L'objectif de notre travail n'est pas de proposer une modélisation des règles ou une alternative aux modélisations existantes. Ainsi, l'ontologie support de notre approche, à savoir l'ontologie e-Compliance, est utilisée telle que présentée par ses auteurs, à quelques ajouts de vocabulaire près.

suivante, nous posons les bases de notre approche de RAQ appliquée à une base de connaissances légales. À notre connaissance, c'est la première approche à s'intéresser à la RAQ dans le domaine des ontologies légales. En plus de s'attaquer aux spécificités du domaine réglementaire, notre approche dispose d'un mécanisme de résolution de décalages en faisant appel à des patrons de questions ce qui permet entre autres la formalisation des questions complexes.

6. Notre approche de RAQ

Considérant le corpus de questions en tant que langage formel, il est tout à fait naturel de procéder à une analyse syntaxique des questions textuelles pour obtenir l'arbre de dérivation correspondant à chaque question, et ensuite traduire cet arbre en requête SPARQL. Or, cette approche est inefficace puisqu'elle mettrait au même niveau les représentations lexicales d'objets de l'ontologie, celles décrivant les propriétés de ces objets, et *last but not least* les mots grammaticaux qui sont indépendants du domaine et ne dépendent que de la langue. Elle demanderait donc l'élaboration d'une grammaire monolithique, où la moindre omission d'un détail entraînerait l'échec de l'analyse.

Au lieu de cela nous procédons d'abord à un *chunking* (une analyse syntaxique superficielle) en identifiant en priorité dans les *chunks* les représentations lexicales d'objets de l'ontologie. Les *chunks* contenant de telles représentations sont alors associés à des sommets intermédiaires de l'arbre de dérivation, et leur ensemble peut donc, le cas échéant, correspondre à un ou plusieurs patrons de questions (§ 6.2). Ensuite on traite un autre type de *chunk*, ceux qui représentent des *descriptions d'entités* (§ 6.3). Un troisième type de *chunk* correspond aux mots grammaticaux qui régissent les questions. Ces trois types de *chunk* participent chacun à sa manière à la traduction de la question en requête SPARQL. Les étapes de notre approche sont présentées sur la figure 4.

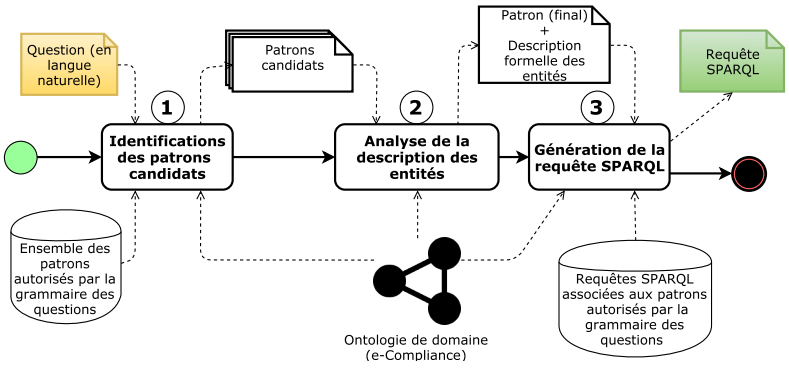


Figure 4. *Étapes de notre approche de RAQ*

6.1. *Description des différents types de questions*

En nous inspirant de travaux de RAQ s'appuyant sur les patrons des questions (paragraphe 4, point 7), nous avons étudié la structure d'un certain nombre de questions de notre domaine d'application. Ces questions, présentées en annexe B, proviennent de la section « Perspectives » du projet e-Compliance[12] et de séances de travail avec les parties prenantes du projet REIZHMOR. En étudiant le corpus de questions, nous leur avons attribué une catégorie en fonction de l'objectif. Ces catégories sont celles de questions : (1) permettant de connaître la valeur d'une propriété ; (2) servant à obtenir la liste des entités ayant une certaine (valeur de) propriété ; (3) permettant de retrouver les réglementations s'appliquant à certaines situations (décrites dans la question) ; (4) permettant de retrouver les textes ou articles qui prescrivent un comportement donné ; (5) servant à identifier les conditions ou les contextes dans lesquels on peut ou non effectuer des actions données ; (6) permettant d'exhiber les entités cibles d'une réglementation ; (7) relatives aux procédures et aux protocoles ; (8) concernant la structure ou les métadonnées des textes réglementaires ; (9) concernant l'application d'un texte ou d'un article.

Environ 63 % des questions du corpus appartiennent aux catégories (1) à (6) ; les 37 % restant se trouvent dans les catégories (7) à (9). Les patrons de questions que nous proposons correspondent aux questions des catégories (1) à (7). Leur extension aux patrons des catégories de questions restantes est un travail en cours. Comme annoncé, les patrons de questions que nous proposons s'appuient sur un langage formel dont nous donnons en annexe A la grammaire régulière dans le formalisme EBNF (*Extended Backus-Naur Form*). Notons que :

– une question est la séquence (cf. ligne 1 du listing) d'un élément de restriction optionnel `Restriction`, d'un groupe de mots codant le type de la question `QuestionType`, du corps de la question `QuestionBody`, et d'un signe d'interrogation `QuestionMark` signifiant sa fin ;

– `Restriction` est un élément qui permet de restreindre l'espace de recherche d'une question à un ensemble de textes ; cela explique le fait qu'il ait la même définition que la référence à un texte réglementaire `ReferenceToReg` (cf. l. 2 du listing). Il fait référence à des expressions telles que : « Dans l'article 2.3 », ou « Dans l'annexe 1 de la Convention internationale pour la prévention de la pollution par les navires », etc. ;

– le corps de la question `QuestionBody`, est formé par l'un des six types de questions que nous avons mentionnés auparavant : `ValueOfProp`, `EntitiesHaving-Prop`, `RulesApplyingToCases`, `RulesPrescribingRequirements`, `Condition-ForActivities` et `TargetedEntities` (cf. l. 9 et 10) ;

– `ValueOfProp` symbolise les questions relatives à la valeur d'une propriété (catégorie (1)). Par exemple « Quelle est la date de publication de la Convention internationale de 1973 ? » ;

12. `http://www.e-Compliance-project.eu/`

– `EntitesHavingProp` est le symbole des questions sur les entités ayant une propriété dont la valeur remplit certaines conditions (catégorie (2)). Par exemple « Quels documents ont été publiés avant 2011 ? » ;

– `RulesApplyingToCases` représente les questions sur les règles s'appliquant dans des cas donnés (catégorie (3)). « Quelles exigences s'appliquent aux engins nautiques ? » en est un exemple ;

– `RulesPrescribingRequirements` permet de détecter les questions sur les règles prescrivant un certain comportement (catégorie (4)). Par exemple, « Quelles exigences interdisent la pêche dans les ports du Morbihan ? » ;

– `ConditionForActivities` symbolise les questions de la catégorie (5) c'est-à-dire relatives aux conditions dans lesquelles peuvent (ou pas) s'exercer certaines activités. Comme exemple de question on a « Dans quels ports de la Manche les pétroliers peuvent-ils mouiller ? » ;

– le symbole `TargetedEntities` est celui de la catégorie des questions qui s'intéressent aux entités *ciblées* par un texte légal donné. Par exemple, nous avons : « Quels types de navires sont concernés par l'arrêté 25/67 du 5 juillet 1967 ? » ;

– dans le corps d'une question, à travers les différents symboles codant les catégories de questions que nous avons relevées, on peut retrouver la description des différentes entités présentes dans la question. Dans le cadre de l'ontologie maritime légale e-Compliance, ces entités peuvent être soit un rôle, soit une organisation, soit un navire. Les descriptions de ces trois entités sont symbolisées respectivement par `RoleDesc`, `OrganisationDesc` et `ShipDesc` ;

– la description d'une cible (*target*), par exemple `ShipDesc`, se compose d'une mention de la cible (*chunk* de premier type) et éventuellement de la description proprement dite (*chunk* de type 2). Cette description constitue un sous-langage formel décrit par l'automate de la figure 5. En guise d'illustration, on peut reconnaître des expressions telles que : « navire marchand », « navire marchand ayant un tonnage supérieur à 400 », « navire sans moteur », etc. ;

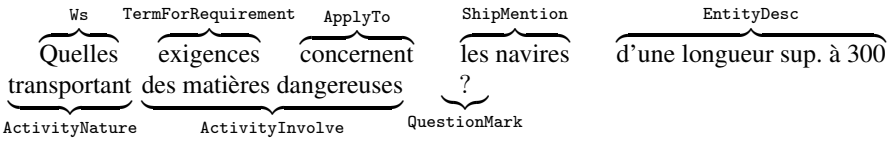
– nous allons voir comment à chaque patron de question, on peut faire correspondre un patron de requêtes SPARQL. Ainsi, le patron `Ws BePredicate TermForRequirement ApplyTo ShipType QuestionMark` a pour patron de requêtes SPARQL :

```
SELECT DISTINCT ?title ?content
WHERE {
        ?rule a :Rule; :isDerivedFromClause ?clause;
                       :hasTarget ?ship.
        ?clause a :Clause; :title ?title; rdfs:label ?content.
        ?ship a :Ship; :type "<ShipType>".}
```

Lors de l'exécution de cette requête, le *slot* `<ShipType>` sera remplacé par l'occurrence de `Shiptype` dans la question en langage naturel.

6.2. *Identification des patrons candidats*

C'est la première étape de notre approche telle que présentée dans la figure 4. Nous effectuons un *chunking* (analyse syntaxique superficielle) de la phrase où chaque *chunk* correspond à la représentation lexicale d'un certain sommet intermédiaire de l'arbre de dérivation de la phrase. Ce *chunking* détecte en priorité les représentations lexicales d'objets (classes, individus, relations) de l'ontologie de domaine (dans notre cas d'application, l'ontologie e-Compliance étendue par nos soins), ainsi que les mots grammaticaux spécifiques à la réalisation syntaxique des questions. Ainsi, la question « Quelles exigences concernent les navires d'une longueur supérieure à 300 transportant des matières dangereuses ? » est découpée de la manière suivante :



où `Ws` (l'adjectif interrogatif « quelles ») est un *chunk* de troisième type, `EntityDesc` est (comme son nom l'indique) un *chunk* de deuxième type, et tous les autres sont des *chunks* de premier type puisqu'ils correspondent à des objets de l'ontologie de domaine.

Au terme de cette étape, nous disposons d'un patron de question. Ce dernier peut contenir des symboles non terminaux représentant des descriptions d'entités qu'il est nécessaire de décoder. C'est le but de la deuxième étape. Dans l'exemple ci-dessus, on a une description d'entités (sommet `EntityDesc`) qui est « longueur supérieure à 300 ». Il faudra donc analyser cette expression.

6.3. *Analyse de la description des entités*

L'étape d'analyse de la description des entités permet de formaliser les fragments de la question qui ne correspondent ni à des concepts, individus ou relations de l'ontologie de domaine, ni à des mots grammaticaux généralement présents dans les questions relatives aux corpus réglementaires (elle correspond à l'étape 2 de la figure 4). Nous considérons, de manière heuristique, que ces fragments servent à apporter un complément d'information aux concepts, individus et relations qui les encadrent. Pour ce faire, nous nous servons d'une version augmentée de l'automate du système CANaLI de (Mazzeo et Zaniolo, 2016) (cf. figure 5). Comme cet automate nécessite la présence d'une entité, nous l'appliquons non pas à `EntityDesc` seul, mais à la paire `OpenEnt EntityDesc` où `OpenEnt` est l'entité qui précède[13] `EntityDesc` (dans notre cas, `ShipMention`, lexicalisée par « les navires »). Nous avons ajouté des transitions à l'automate permettant un spectre plus large de descriptions. Par exemple, des descriptions telles que « Navire de 5 (mètres) de long » ou « Navire long de plus de 5 (mètres) » ne sont pas reconnues par l'automate original alors que la version augmentée que nous proposons les reconnaît.

13. On peut, de manière similaire, étendre l'automate au cas où l'entité suit sa description.

En outre, ce processus doit tenir compte des variantes lexicales possibles des termes de `EntityDesc`, appelées *lexicalisations* dans le cadre du projet DBpedia (Mendes *et al.*, 2012). Ainsi, le mot « long » peut être associé à des propriétés de l'ontologie de domaine impliquant une longueur, et dans l'ontologie e-Compliance, l'adjectif « long » peut être associé à `:maxLength` et `:minLength` qui sont des propriétés de la classe `:Ship`. L'analyse de la description d'entités se décompose alors en deux sous-étapes : lexicalisation et reconnaissance par l'automate.

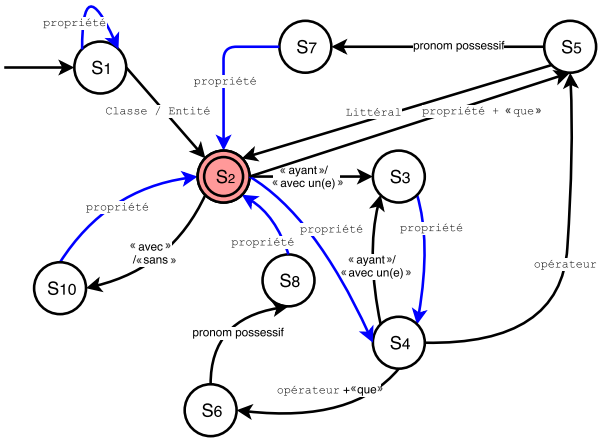


Figure 5. *Automate pour la reconnaissance d'une expression décrivant une entité (adapté de Mazzeo et Zaniolo (2016))*

6.3.1. *Lexicalisation*

Nous procédons à un *chunking* du contenu textuel de la description d'entités. Dans chacun des *chunks*, on procède à la recherche des variantes lexicales possibles. Pour cela nous nous servons les formes lexicalisées des classes[14] et des propriétés[15] de DBpedia (Unger *et al.*, 2013). Nous disposons ainsi de la version en langage naturel et des éventuelles variantes de chaque *chunk*. Nous enrichissons cet ensemble, fondé sur DBpedia, en lui ajoutant les représentations lexicales des propriétés de notre ontologie de domaine, obtenues par proximité au niveau des chaînes de caractères, par la distance de Levenshtein.

Nous illustrons cette étape avec la description « d'une longueur supérieure à 300 » qu'il nous reste à décoder après l'étape d'identification des patrons candidats (voir section 6.2). Une analyse syntaxique superficielle de ce fragment nous donne les deux

14. `https://github.com/dice-group/hawk/blob/master/resources/dbpedia_3Eng_class.ttl`
15. `https://github.com/dice-group/hawk/blob/master/resources/dbpedia_3Eng_property.ttl`

chunks « d'une longueur » et « supérieure à 300 ». Les traitements effectués à l'aide de ces deux *chunks* sont regroupés dans le tableau 1.

	« d'une longueur »	« supérieure à 300 »
Ajout par lexicalisation	*length*	-
Ajout par alignement avec l'ontologie	`:maxLength`, `:minLength`	-
Interprétations possibles de « d'une longueur supérieure à 300 »	(1) « d'une longueur supérieure à 300 » (2) « *length* supérieure à 300 » (3) « `:maxLength` supérieure à 300 » (4) « `:minLength` supérieure à 300 »	

Tableau 1. *Obtention des variations lexicales possibles du fragment « d'une longueur supérieure à 300 »*

6.3.2. *Analyse de la description d'entités*

Au cours de cette sous-étape, nous filtrons les variantes lexicales possibles d'une description d'entités en ne gardant que celles qui sont reconnues par l'automate. Notons que nous gardons les états et les transitions des variantes lexicales qui sont reconnues par l'automate puisqu'elles nous serviront à la formalisation proprement dite de la description. Nous synthétisons les résultats de ce processus pour notre exemple, dans le tableau 2. Dans ce tableau, on a d'un côté les interprétations candidates et de l'autre les états et transitions obtenus lors des tentatives de reconnaissance de ces phrases par l'automate. Ajouter à la description proprement dite l'entité qui précède nous permet de confirmer que la description correspond effectivement à l'entité. En effet, dire qu'une description est celle d'une entité impose de devoir faire *une validation sémantique en plus d'une validation syntaxique*. Dans ce cas, la validation syntaxique repose sur les états et transitions de l'automate et la validation sémantique s'effectue en s'assurant que :

Interprétations	*États et transitions*
(1) les navires + d'une longueur supérieure à 300	$S_1 \rightarrow S_2 \rightarrow$ X
(2) les navires + *length* supérieure à 300	$S_1 \rightarrow S_2 \rightarrow$ X
(3) les navires + `:maxLength` supérieure à 300	$S_1 \rightarrow S_2 \rightarrow S_4 \rightarrow S_5 \rightarrow S_2$
(4) les navires + `:minLength` supérieure à 300	$S_1 \rightarrow S_2 \rightarrow S_4 \rightarrow S_5 \rightarrow S_2$

Tableau 2. *États et transitions pour le fragment « d'une longueur supérieure à 300 ». Le symbole* X *signifie un échec de la reconnaissance de la phrase par l'automate*

– lorsqu'on rattache une propriété à une classe alors cette dernière est incluse dans le *domaine*, au sens de la propriété `rdfs:domain`[16], de cette propriété ;

– lorsqu'un comparateur (*e.g.* « inférieur à », « plus grand que ») suit une propriété, alors cette propriété est de type comparable (*e.g.* entier, flottant, date, etc.). De même,

16. Le préfixe `rdfs` fait référence à `https://www.w3.org/2000/01/rdf-schema`

lorsqu'une (valeur de) propriété est comparée à un littéral, on vérifie que ce dernier est de type compatible à celui de la propriété. On évitera ainsi, de valider la comparaison d'un flottant et d'une date.

Ainsi pour les lignes (1) à (4) du tableau 2, on cherche à valider syntaxiquement et sémantiquement l'analyse de la description « d'une longueur supérieure à 300 » rattachée à l'entité (`OpeningEnt`) « les navires ». Rappelons que cette entité a été décodée en tant que référence à la classe `ecom:Ship` de l'ontologie de domaine. Des quatre interprétations candidates, seules la (3) et la (4) sont reconnues avec :

– entre les états S_1 et S_2 une transition avec l'entrée « les navires » qui fait référence à la classe `:Ship` ;

– entre S_2 et S_3 une transition avec l'entrée `:maxLength` (resp. `:minLength`) pour l'interprétation (3) (resp. (4)). Cette transition est aussi validée sémantiquement, car la classe `:Ship` en est le domaine des propriétés `:maxLength` et `:minLength` ;

– entre S_4 et S_5 il y a le comparateur « supérieur à » qui assure la transition. De plus il est sémantiquement compatible avec le domaine d'arrivée de `:maxLength` et `:minLength` ;

– entre S_5 et S_2 on a le littéral « 300 » qui, en outre, est en adéquation avec le domaine d'arrivée de `:maxLength` et `:minLength`.

Notons que dans cet exemple les interprétations (3) et (4) sont toutes deux reconnues par l'automate. Nous choisissons l'interprétation finale de manière aléatoire parmi ces deux interprétations.

Au terme de cette étape, nous disposons :

1) du patron de la question obtenu dès la première étape de notre approche (section 6.2) ;

2) des interprétations des descriptions d'entités du patron et des états et transitions ayant permis de valider chacune de ces interprétations.

Nous pouvons désormais générer la requête qui nous permettra d'interroger la base de connaissances réglementaires et donc de répondre à la question posée.

6.4. *Génération de la requête SPARQL*

Notre base de connaissances est composée de règles qui constituent des représentations formelles des règles textuelles provenant des arrêtés préfectoraux. Toute règle de la base de connaissances est une instance de la classe `:Rule` de l'ontologie e-Compliance. Rappelons que e-Compliance nous sert à la fois d'ontologie de domaine et d'ontologie légale. En tant qu'ontologie de domaine, et donc ontologie maritime, elle dispose de structures pour représenter le domaine maritime (les navires, les activités, les rôles et organisations, les juridictions, etc.). En tant qu'ontologie légale elle propose une modélisation ontologique des *règles* (une règle ayant un contexte, une cible et une exigence) (Lohrmann *et al.*, 2014).

Le formalisme retenu pour e-Compliance étant le langage OWL[17] (*Web Ontology Language*), le moyen adéquat pour l'interroger est de se servir du langage de requêtes standard SPARQL[18] (*SPARQL protocol and Query Language*). Ce dernier permet d'interroger des données RDF, RDFS, OWL. Il propose quatre types de requêtes :

– `SELECT` pour sélectionner tous les éléments répondant à des critères donnés ;

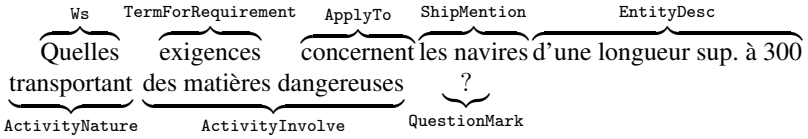
– `ASK` pour affirmer ou infirmer l'existence d'éléments répondant à des critères donnés ;

– `DESCRIBE` pour obtenir une description d'un certain nombre de ressources ;

– `CONSTRUCT` pour construire de nouvelles ressources à partir d'un ensemble d'entités répondant à des critères donnés.

Pour notre approche de RAQ, seules les requêtes SPARQL de type `SELECT` et `ASK` sont pertinentes. En effet, soit il sera question d'identifier et de retourner les éléments de réponse à une question, soit il faudra répondre si oui ou non, le cas décrit dans une question est prévu dans la base de connaissances. De même, dans la grammaire EBNF des questions (annexe A p. 69), à la ligne 3, nous prévoyons deux types de questions, symbolisés par `Select` et `Ask`.

Rappelons qu'à ce stade nous disposons du patron de la question et d'éléments de formalisation des descriptions éventuelles d'entités présentes dans le patron. Aussi rappelons qu'à chaque patron de question il y a une requête SPARQL associée. L'annexe C donne des exemples de patrons ainsi que les requêtes associées. Cet état de fait est illustré par l'étape 3 de la figure 4. Ainsi pour notre exemple, il nous faut instancier la requête SPARQL correspondant au patron de la requête. Le patron de notre question est :



La requête SPARQL associée à ce patron est :

```
1  SELECT DISTINCT ?title
2  WHERE {
3          ?rule rdf:type :Rule; :isDerivedFromClause ?clause;
4                        :hasRequirement ?req;
5                        :hasTarget ?ship. ?ship rdf:type :Ship.
6          ?clause rdf:type :Clause.
7          ?clause :isPartOf ?reg.
8          ?reg rdf:type :Regulation; :title ?title.
9          ?req :nature "transport"^^xsd:string.
10         ?req :involves ?cargo.
```

17. https ://www.w3.org/OWL/

18. https ://www.w3.org/TR/rdf-sparql-query/

```
11          ?cargo rdf:type :Cargo; :isDangerous "true"^^xsd:boolean.
12          [?ship :maxLength ?length. FILTER(?length > 300)]}
```

Cette requête a été obtenue en instanciant le patron correspondant (voir annexe C) avec :

– la classe `:Ship` comme cible (propriété `:hasTarget`) des règles pertinentes pour notre question (voir ligne 5 de la requête) ;

– la nature des activités, `"transport"` (voir ligne 9), ainsi que la nature des éléments impliqués dans ce transport à savoir une cargaison de produit dangereux (voir ligne 11) ;

– la description de la cible visée par les règles pertinentes pour la question. Cette description est donnée par la ligne 12 de la requête.

Comme nous l'avons mentionné dans la section 5.2, à cause du décalage d'articulation ontologique important entre les ontologies légales et les règles en langue naturelle, les requêtes SPARQL peuvent être relativement complexes. Le fait de disposer pour chaque patron de question du patron de la requête SPARQL correspondante permet d'appréhender ce phénomène.

7. Évaluation

Nous avons évalué la capacité de formalisation de notre approche sur un corpus constitué des questions mentionnées en annexe B. Ce corpus est constitué de trente questions en langage naturel, ciblant les textes du corpus réglementaire décrit à la section 2. Sur ces trente questions :

1) quatorze questions, soit 46,67 %, sont correctement formalisées. Les patrons correspondant à ces questions sont convenablement identifiés et instanciés ;

2) les patrons de cinq questions, soit 16,67 %, ne sont pas identifiés. Bien que ces questions appartiennent aux six catégories de questions que nous avons identifiées et pour lesquelles des patrons sont proposés, notre langage formel ne couvre pas la syntaxe des questions ;

3) onze questions, soit 36,67 %, ne sont pas identifiées, (i) soit parce que le type de la question n'est pas pris en compte dans la syntaxe (exemple : la question 27 sur le nombre d'articles de l'arrêté 96/2015), (ii) soit parce que l'ontologie maritime e-Compliance, qui sert de support à notre base de connaissances, ne formalise pas les informations nécessaires pour répondre à la question (exemple : la question 30 sur les informations à fournir en cas de découverte d'engin suspect).

Cette première évaluation montre qu'une extension de la syntaxe ainsi que de l'ontologie – à la fois légale et maritime – e-Compliance sont à considérer dans nos travaux futurs, que nous présentons dans la section suivante. Notons toutefois que la métrique de notre évaluation doit aller au-delà du regard de la qualité de la résolution du patron d'une question. En effet, *un patron peut être bien identifié mais incorrectement instancié*. Cela arrive principalement avec la formalisation des descriptions des entités. On peut le voir dans l'exemple que nous avons déroulé pour illustrer notre approche.

Dans cet exemple, nous avons formalisé l'expression « d'une longueur supérieure à 300 » (voir tableaux 1 et 2). À l'issue de la formalisation de cette expression le terme « longueur » est formalisé par la propriété `:maxLength`. Or d'après les experts, le prédicat correct dans ce cas est `:minLength`. Disposer des validations des experts ainsi que des réponses précises aux différentes questions du corpus est un travail à faire pour améliorer notre méthode d'évaluation.

8. Travaux futurs

L'approche de RAQ que nous avons présentée fait apparaître les points d'amélioration que voici :

– *l'extension de la portée des types de questions et de leur syntaxe.* Parmi les questions qui sont à la portée de notre approche, on retrouve essentiellement celles qui sont liées au contenu des réglementations : ce qu'elles prescrivent, dans quel contexte, etc. Ce sont, par exemple, les questions relatives à la structure des textes réglementaires, ou encore liées aux relations entre les textes ou aux procédures et informations mentionnées dedans ;

– *l'extension de l'ontologie maritime e-Compliance.* Nous avons relevé le fait que l'ontologie e-Compliance, dans sa forme actuelle, n'est pas capable de représenter certaines informations contenues dans les réglementations, diminuant de ce fait la performance des approches de RAQ. Une extension de cette ontologie est nécessaire pour prendre en compte la représentation de la structure des textes réglementaires ainsi que les relations hiérarchiques entre les textes, et pour représenter plus en détail le contenu des textes (par exemple les procédures à suivre). En outre il faut augmenter la représentation des règles de cette ontologie de manière à avoir une règle au sens classique du terme, *i.e.* `antécédent` $\Longrightarrow$ `conséquent`. Cela permettrait de disposer d'un dépôt de règles formelles et d'effectuer un contrôle automatique de conformité (Yurchyshyna et Zarli, 2009 ; Kacfah Emani, 2016) ;

– *proposition d'une approche plus générale.* En nous appuyant sur l'approche de RAQ présentée, nous envisageons de proposer une méthodologie générale de RAQ vis-à-vis d'une base de connaissances légales, le cas de la réglementation maritime ne constituant qu'un cas d'application. Pour démontrer la pertinence de cette méthodologie, il faudra l'appliquer à d'autres domaines d'application ;

– *mise sur pied d'un corpus d'évaluation.* Il est nécessaire de disposer de bancs d'essai pour évaluer les approches de RAQ sur des bases de connaissances légales. De tels bancs d'essai doivent comprendre : (i) plusieurs ontologies légales multilingues – une fois peuplées, ces ontologies serviraient de bases de connaissances cibles pour les questions en langage naturel ; (ii) un nombre important de questions en langage naturel avec les réponses attendues, ainsi que les requêtes SPARQL correspondantes, validées par des experts du domaine. Ces questions doivent couvrir la diversité des types de requêtes que les utilisateurs usuels des bases de connaissances légales se posent.

9. Conclusion

Nous avons présenté les premières briques du projet REIZHMOR dont le champ d'application est celui de la réglementation maritime. Nous avons décrit le premier corpus de ce projet qui est composé d'arrêtés préfectoraux et interpréfectoraux ; au cours de l'évolution du projet, ce corpus devra être étendu à des textes de niveau national et international afin de proposer des preuves de concept robustes. Nous avons avancé une approche de réponses automatiques aux questions (RAQ) en langage naturel dans le domaine légal appliqué à la réglementation maritime. En plus d'aborder les problèmes généraux de RAQ tels que la variété lexicale, elle propose des solutions pour des problèmes spécifiques au domaine légal, telles que la résolution du décalage d'articulation ontologique à l'aide de patrons de questions. Une première évaluation de notre approche montre des résultats prometteurs avec plus de 46 % des questions correctement formalisées. Pour améliorer ces résultats, nous prévoyons d'étendre le niveau d'informations formalisées par les ontologies maritimes existantes et aussi la portée des patrons syntaxiques des questions. En outre, nous visons à proposer une méthodologie générale de RAQ pour les bases de connaissances légales.

Remerciements

Ce travail est financé par le Service hydrographique et océanographique de la marine (Shom) dans le cadre du projet REIZHMOR.

10. Bibliographie

Berners-Lee T., Hendler J., Lassila O. *et al.*, « The semantic Web », *Scientific American*, vol. 284, n° 5, p. 28-37, 2001.

Cornu G., *Linguistique juridique*, Montchrestien, Paris, 2005.

Daille B., « Conceptual structuring through term variations », *in* F. Bond, A. Korhonen, D. MacCarthy, A. Villacicencio (eds), *Proceedings ACL 2003 Workshop on Multiword Expressions : Analysis, Acquisition and Treatment*, p. 9-16, 2003.

de Cet Bertin C., *Introduction au droit maritime*, ellipses, Paris, 2008.

Haralambous Y., Sauvage-Vincent J., Puentes J., « A Hybrid (Visual/Natural) Controlled Language », *Languages Resources and Evaluation*, vol. 51, n° 1, p. 93-129, 2017.

Hirschman L., Gaizauskas R., « Natural language question answering : the view from here », *Natural Language Engineering*, vol. 7, n° 4, p. 275-300, 2001.

Höffner K., Walter S., Marx E., Usbeck R., Lehmann J., Ngonga Ngomo A.-C., « Survey on challenges of Question Answering in the semantic Web », *Semantic Web*, vol. 9, p. 1-26, 2016.

Kacfah Emani C., Formalisation automatique et sémantique de règles métiers, thèse de doctorat, Université de Lyon, 2016.

Kuhn T., « A survey and classification of controlled natural languages », *Computational Linguistics*, vol. 40, p. 121-170, 2014.

Lohrmann P., Seizou M., Hagaseth M., Griffiths D., A European Maritime e-Compliance Cooperation Model - Ontology, Technical Report n° 2.2, Seventh Framework Program, 2014.

Lopez V., Uren V., Sabou M., Motta E., « Is question answering fit for the semantic Web ? : A survey », *Semantic Web*, vol. 2, n° 2, p. 125-155, 2011.

Massachusetts Senate, *Legislative Drafting and Legal Manual*, 2010. `https://malegislature.gov/Content/Documents/General/LegislativeDraftingManual.pdf`.

Mazzeo G. M., Zaniolo C., CANaLI : A System for Answering Controlled Natural Language Questions on RDF Knowledge Bases, Technical Report n° 160004, 2016. `http://fmdb.cs.ucla.edu/Treports/canali_tr_160004.pdf`.

Mendes P. N., Jakob M., Bizer C., « DBpedia - A Multilingual Cross-domain Knowledge Base », *Proceedings of LREC 2012, Eighth International Conference on Language Resources and Evaluation*, p. 183-1817, 2012.

Nivre J., Hall J., Nilsson J., « MaltParser : A Data-Driven Parser-Generator for Dependency Parsing », *Proceedings of the fifth international conference on Language Resources and Evaluation (LREC 2006), Genoa, Italy*, p. 2216-2219, 2006.

Sauvage-Vincent J., Un langage contrôlé pour les *Instructions nautiques* du Service Hydrographique et Océanographique de la Marine, thèse de doctorat, IMT Atlantique, 2017.

Shom, *France (côtes Nord et Ouest). De la frontière belge à la pointe de Penmarc'h*, vol. C2A of *Instructions nautiques*, 2010. Édition à jour le 21 juin 2017.

Simperl E., Tempich C., Vrandečić D., « A methodology for ontology learning », *in* P. Buitelaar, P. Cimiano (eds), *Ontology learning and population : bridging the gap between text and knowledge*, IOS Press, p. 225-249, 2008.

Unger C., McCrae J., Walter S., Winter S., Cimiano P., « A lemon lexicon for DBpedia », *Proceedings of the 2013 International Conference on NLP & DBpedia-Volume 1064*, CEUR-WS.org, p. 103-108, 2013.

Yurchyshyna A., Zarli A., « An ontology-based approach for formalisation and semantic organisation of conformance requirements in construction », *Automation in Construction*, vol. 18, n° 8, p. 1084-1098, 2009.

Annexes

A. Extrait de la syntaxe EBNF des questions adressées à une base de connaissances réglementaires (appliquée au domaine maritime)

Les numéros ① à ⑥ indiquent les symboles représentant les catégories de questions (1) à (6) présentées à la section 6.1.

```
Question = Restriction?, QuestionType, QuestionBody, QuestionMark;
Restriction = ReferenceToReg;
QuestionType = Select | Ask;
Select = Ws, BePredicate;
Ws = ("Qui"|"Quel"|"Dans quel");
BePredicate = ('est' | 'sont');
Ask = ('Est-ce que');
QuestionMark = '?';
QuestionBody = ValueOfProp | EnitiesHavingProp | RegApplyingToCases
| RulesPrescribingRequirements | ConditionForActivities | TargetedEntities;
```

```
① ValueOfProp = (Property, Of)?,  (Class | Individual);
② EntitiesHavingProp = (Class|Entity), Property, Comparator, Value;
③ RegApplyingToCases = ReferenceToReg, ApplyTo, Case;
Case = TargetCase | ActivityCase;
TargetCase = (TargetDesc, (Perform, Activity)?) | (Actity, Of, TargetDesc);
TargetDesc = TargetMention, Description?;
ActitityCase = Activity;
TargetMention = ShipMention | RoleMention | OrganisationMention;
Activity = ActivityNature, Invole?, Place?, Situation?;
Place = Jurisdiction?;
Situation = MaritimeSituation?;
④ RegPrescribingRequirements = RegHavingModality | PrescriptionOnEntity;
RegHavingModality = ReferenceToReg, BePredicate, Modality;
PrescriptionOnEntity = ReferenceToReg?, ApplyTo?, ModalityOnEntity, Requirement?;
ModaliyOnEntity = (Modality, EntityDesc)| (EntityDesc, Modality);
EntityDesc = TargetDesc | Activity;
Requirement = Activity | Doc | Equip | Role;
⑤ TargetedEntities = TargetAndType, PredicateForTarget, In, ReferenceToReg;
⑥ ConditionForActivities = (Condition, ModalityOnActivity)
| (Condition, ModalityOnEntity, TargetCase) | (Condition, Activity, ModalityonEntity);
ModalityOnActivity = Modality, Activity | Activity, Modality;
Condition = (Jurisdiction | MaritimeSituation);
ReferenceToReg = TypedRegulation | TermForRequirement;
TypedRegulation = "reglementation", (In, DocName)? | Qualifier, "regulation";
DocName = ... Name of a Legal Text ... ;
TermForRequirement = "exigence", (In, DocName)?
TargetDesc = ShipDesc | RoleDesc | OrganisationDesc;
ShipDesc = ShipMention, DescriptionOfShipItself?;
ShipMention = ShipType | SynonymOfShip;
```

B. Questions du corpus de test

① à ⑥ : catégories de questions dont les patrons sont correctement identifiés, ⑦ à ⑫ : catégories de questions dont les patrons sont correctement identifiables après extension de syntaxe, ⑬ à ㉑ : autres cas.

	Questions	**Catégories**
1	Quelle est la définition de vraquier ?	①
2	Quel est le terme officiel pour désigner un navire utilisé pour la pêche ?	⑦
3	Quelles exigences s'appliquent aux caboteurs de plus de 60 mètres de long ?	⑨
4	Quels textes se rapportent aux obligations d'un capitaine de port ?	④
5	Quelles sont les règles relatives à la maintenance des navires à grue ?	③
6	Quels arrêtés préfectoraux sont relatifs à la sécurité au port de Brest ?	③
7	Le mouillage est-il autorisé dans le goulet de Brest ?	④
8	Quels sont les textes qui réglementent l'accès au port de Port-en-Bessin ?	④
9	La navigation est-elle autorisée au large de la digue du Break pour les navires de pêche ?	④
10	Quelles autorités sont chargées de l'application de l'arrêté n° 22/91 ?	㉑
11	La maintenance des navires à cargaison sèche est-elle autorisée au port de Brest ?	④

12	Peut-on pratiquer la plongée sous-marine à 200 mètres du sablier TIMAC ?	⑯
13	Quelles sont les poursuites prévues par l'arrêté nº 143/92 ?	⑲
14	Dans quels ports de France un navire de charge à pont ouvert peut-il accoster ?	⑫
15	Sous quelles conditions météorologiques un transbordement en mer du Nord est-il autorisé ?	⑥
16	Quels sont les visas de l'arrêté nº 61/94 ?	⑳
17	Quelles activités sont interdites dans la pointe de la Torche ?	④
18	Quelles sont les conditions pour une demande de dérogation à l'interdiction de stationnement au port de Saint-Malo ?	⑲
19	Quels articles s'appliquent aux navires transportant des hydrocarbures ou des substances dangereuses ?	⑨
20	Quels navires sont concernés par l'arrêté interpréfectoral nº 2002/99 Brest 2002/58 Cherbourg ?	⑤
21	Quelles sont les zones concernées par l'arrêté nº 2009/55 ?	⑤
22	L'arrêté nº 22/2009 est-il encore en vigueur ?	⑳
23	En cas de naufrage, peut-on mouiller dans un centre nucléaire ?	⑥
24	Quelle est la procédure d'autorisation pour pratiquer des activités sportives au large du centre nucléaire de la Penly ?	⑲
25	Quelles sont les limites du port civil de Cherbourg ?	⑬
26	Quelle est la vitesse maximale autorisée dans la petite rade de Cherbourg ?	⑬
27	Combien d'articles compte l'arrêté 96/2015 ?	⑳
28	Est-il interdit de pratiquer des activités nautiques dans l'anse du Poulmic ?	⑥
29	Qu'est-ce qu'un engin suspect ?	①
30	Quelles informations communiquer en cas de découverte d'un engin suspect ?	⑲

Rappelons qu'à la section 6.1, nous avons classé les questions en neuf catégories. Pour les questions des catégories 1 à 6 nous avons proposé des patrons (voir les symboles numérotés ① à ⑥ en annexe A). Nous mentionnons ci-dessus que les étiquettes ⑬ à ㉑ font référence aux « autres cas ». Parmi ces cas :

– les étiquettes ⑬ à ⑱ font référence aux questions des catégories 1 à 6. Cependant, les patrons proposés ne prennent pas en compte la syntaxe de ces questions et dans le même temps, le schéma de l'ontologie de domaine support, e-Compliance, ne permet pas de représenter les éléments de réponse pour cette question. Pour illustration, considérons la question 12 « Peut-on pratiquer la plongée sous-marine à 200 mètres du sablier TIMAC ? ». Pour répondre à cette dernière, il faudrait que la juridiction de déroulement d'une activité permette d'effectuer les calculs de distance adéquats. Ceci n'est pas le cas dans la version actuelle d'e-Compliance. Comme autre exemple, on peut prendre la question 26 « Quelle est la vitesse maximale autorisée dans la petite rade de Cherbourg ? ». Pour pouvoir y apporter des éléments de réponse il faut associer à chaque juridiction un certain nombre de paramètres, dont la vitesse. De plus la valeur de ces paramètres peut être contextuelle (par exemple : vitesse en cas d'intempérie, vitesse en fonction de la saison, de l'affluence, etc.) ;

– les étiquettes ⑲ à ㉑ font référence aux questions des catégories 7 à 9. Nous n'avons pas proposé de patron pour les questions de cette catégorie, elles sont donc hors de la portée de notre approche dans sa version actuelle.

Dans les lignes suivantes, nous commentons quelques-unes de ces questions :

Question 1 : Quelle (Ws) est (Be) la définition (Property) de (Of) vraquier (Entity) ? (?)

C'est une question de catégorie 1, c'est-à-dire permettant de connaître la valeur d'une propriété d'une entité.

Question 2 : Quel (Ws) est (Be) le terme officiel (Property) pour désigner (Of) un navire utilisé pour la pêche (Entity) ? (?)

C'est aussi une question de la catégorie 1, mais elle n'est pas bien identifiée par notre approche. En effet, la syntaxe actuelle des patrons de questions ne reconnaît pas l'expression « pour désigner » comme instance du symbole `Of` servant de liaison entre la propriété et son entité. Aussi, dans cet exemple, l'entité dont on questionne la valeur de propriété n'en est pas vraiment une, mais plutôt une description d'entités.

Question 4 : Quels (Ws) textes (ReferenceToReg) se rapportent (ApplyTo) aux obligations (Modality) d'un capitaine de port (RoleMention) ? (?)

C'est une question de catégorie 4, c'est-à-dire relative à des prescriptions de comportement.

Question 10 : « Quelles autorités sont chargées de l'application de l'arrêté n° 22/91 ? »

Cette question est relative à l'application d'un texte et est classée en catégorie 9. L'ontologie légale et de domaine e-Compliance ne modélise pas ce type d'information. L'extension de l'ontologie que nous prévoyons d'élaborer adressera ce type de cas.

Question 16 : « Quels sont les visas de l'arrêté n° 61/94 ? »

Cette question est de catégorie 8. Elle concerne la structure des textes réglementaires. Une extension de notre ontologie support permettra de représenter et donc d'interroger ce type d'information.

Question 19 : « Quels articles s'appliquent aux navires transportant des hydrocarbures ou des substances dangereuses ? »

Cette question est de catégorie 3. Autrement dit, elle concerne les réglementations s'appliquant à certaines situations. Cependant la syntaxe des patrons de questions ne gère pas de liste de situations. En effet, cette question mentionne deux cas alternatifs : le « transport d'hydrocarbures » et le « transport de substances dangereuses ». Une extension de la syntaxe s'attaquera à ces cas.

C. Exemples de requêtes SPARQL associées aux patrons de questions

Cette ressource est disponible en ligne à l'adresse `http://perso.telecom-bretagne.eu/yannisharalambous/data/tal-58-2-annexe-C.pdf`.

Notes de lecture

Rubrique préparée par Denis Maurel

Université François Rabelais Tours, LI (Laboratoire d'informatique)

Philip WILLIAMS, Rico SENNRICH, Matt POST, Philipp KOEHN. Syntax-based Statistical Machine Translation. *Morgan & Claypool publishers*. 2016. 190 pages. ISBN 978-1-62705-900-8.

Lu par **Fabrice LEFÈVRE**

Université d'Avignon / LIA-CERI

Contrat non rempli. Un ouvrage de bonne qualité, rigoureux, mais qui n'offre pas réellement une synthèse du sujet.

Contrat non rempli. Les *Synthesis on Human Language Technologies*» de Morgan & Claypool commencent à former une belle collection (trente-cinq ouvrages publiés jusqu'à présent), très représentative du domaine du traitement de la langue, diverse, mais aussi généralement fidèle à son appellation. Or la *synthesis* est ce qui fait défaut ici à un ouvrage qui a par ailleurs de grandes qualités. Trop ancré dans sa vision algorithmique, trop pointu, le livre y perd en vision globale synthétique sur le sujet.

La structure de l'ouvrage repose sur quatre parties principales. Un premier chapitre introduit l'ouvrage en s'attachant surtout à présenter les modèles utilisés pour la traduction guidée par la syntaxe. Suit un chapitre entier (chap. 2) consacré à la présentation des trois techniques principales d'extraction de grammaire (Hiero, SAMT, GHKM) à partir de corpus parallèles alignés ou pas. Enfin, élément central de l'ouvrage, le décodage est détaillé dans trois chapitres. Le premier permet de poser les bases d'un formalisme (autour de la notion d'hypergraphe) et rend compte des algorithmes associés (chap. 3). Un chapitre entier est ensuite dédié d'abord au décodage d'arbres syntaxiques (chap. 4.), puis au décodage de chaînes (chap. 5.), qui passent en revue les détails des techniques permettant la mise en œuvre de ces approches. Cette partie importante (77 pages) est suivie d'une partie pêle-mêle qui traite de sujets épars (transformation d'arbres, analyse en dépendance, grammaticalité ou encore le problème de l'évaluation) en une vingtaine de pages (chap. 6). Ces quatre parties sont suivies d'une très (trop) brève section de remarques conclusives (4 pages).

On aura noté donc que dans son développement le livre passe en revue de manière très précise et poussée les techniques de décodage pour une traduction aidée par la syntaxe, y consacrant presque trois chapitres entiers. Pour être honnête, à moins d'être en phase de mise au point d'un système, le livre tombe souvent des

mains tant nombre d'éléments relèvent plus des annexes que du corps principal de la présentation. D'ailleurs les auteurs nous mettent en garde dès la préface où ils qualifient leur livre de *solid fundation for beginning experimental work*. On soulignera malgré tout au passage l'intérêt de l'introduction de la notion d'hypergraphes qui permet de présenter ensuite de manière très cohérente et élégante les différentes instances d'algorithmes de décodage.

Au niveau général du livre, il est patent que l'intérêt d'une traduction guidée par la syntaxe n'est jamais vraiment démontré. C'est un postulat. Mais qui aurait pu, et dû, être appuyé par une analyse plus fine des différences linguistiques entre paires de langues. De même les caractéristiques structurelles (VSO *vs* SVO…) ne sont citées qu'une (ou deux) fois (et pas commentées). À cet égard, il est révélateur que la problématique du réordonnancement ne soit même pas présentée et discutée alors qu'elle représente un des problèmes clés de la traduction que la syntaxe est censée aider.

Encore à titre d'illustration, on pourra noter que l'étiquetage syntaxique, pourtant élément pivot de tout l'exposé, n'est jamais clairement introduit ni explicité (en dépit d'un bref et laborieux retour en fin d'ouvrage). Par exemple, les étiquettes elles-mêmes, pourtant utilisées à profusion dans le texte, ne sont pas définies. Bien sûr, ce manque ne posera guère de difficultés à un spécialiste endurci du TALN, mais fera défaut à un novice cherchant à élargir ses compétences.

Pas de discussion non plus vraiment sur la problématique de l'évaluation. Alors même que dès la préface, il est bien admis que l'intérêt de la syntaxe pour la traduction ne se révèle en général pas autant dans les mesures automatiques classiques, à la BLEU, qu'avec des évaluations humaines. Il aurait été souhaitable que quelques conclusions soient tirées de cela. Car sinon comment justifier tant d'efforts dans une approche dont les bénéfices sont condamnés à ne pas (ou difficilement) être démontrables en pratique ?

Dans la même ligne, on regrettera qu'une plus grande part ne soit pas faite à l'implication des nouvelles approches fondées sur les réseaux de neurones et l'impact qu'elles pourront avoir sur la traduction aidée par la syntaxe. Et ce, alors même qu'un certain nombre de travaux sont déjà engagés sur le sujet (par exemple sur la base de Delvin *et al.*, ACL, 2014). Aussi, alors que la notion de grammaire est bien présentée, dans un sens très large, le rôle de la sémantique n'est pas du tout abordé (relégué comme pour les réseaux de neurones à un très court chapitre dans la partie « Et ensuite ? » de la conclusion). Pourtant une quantité non négligeable de travaux, tels ceux menés à Hong Kong par D. Wu, présentent des résultats intéressants et sont très connectés à la traduction guidée par la syntaxe, notamment en partageant un grand nombre de problématiques (étiquetage, représentation structurée, complexité du décodage…). C'était un choix éditorial possible, mais qui concourt à renforcer l'étroitesse de vue du livre.

Dans un autre registre, il faut noter que la qualité littéraire de l'ouvrage est indéniable. Le texte est agréable à lire dans un anglais parfois très insulaire, mais qui nous sort agréablement de nos habituelles formules technico-scientifiques si formatées, sans perdre de sa précision ni entamer notre compréhension.

En résumé, il s'agit d'un ouvrage scientifiquement très solide, bien étayé par une bibliographie également solide, mais qui semble rater la cible de la série dans laquelle il s'inscrit. Un ouvrage à réserver donc aux lecteurs désireux d'acquérir rapidement les techniques liées à la mise en œuvre d'un système de traduction guidé par la syntaxe sans trop s'inquiéter des raisons qui les y ont conduits et justifient cette option. Les autres devront se tourner vers des présentations plus anciennes, comme la trentaine de pages qu'y consacre le livre de référence d'un des coauteurs, P. Koehn, « *Statistical Machine Translation* ».

Céline POUDAT, Frédéric LANDRAGIN. Explorer un corpus textuel. Méthodes – pratiques – outils. *De Boeck*. 2017. 240 pages. ISBN 978-2-80730-563-2.

Lu par **Chantal ENGUEHARD**

Université de Nantes – LS2N

Cet ouvrage synthétise l'expérience d'un groupe de chercheurs en linguistique en ce qui concerne la méthodologie d'exploitation d'un corpus. Il présente l'originalité d'avoir été construit à partir des pratiques et de répondre à un objectif pédagogique bien identifié : il s'agit de connaître un ensemble d'outils afin d'effectuer un choix fondé sur leurs fonctionnalités, sur les méthodes mises en œuvre, ainsi que les interprétations qui en découleront.

Les auteurs distinguent deux méthodologies principales d'exploration de corpus. L'analyse peut être fondée sur le corpus. Il s'agit alors d'une démarche déductive dans laquelle les données constituent un appui à une théorie linguistique. Ou bien l'analyse peut être guidée par le corpus dans une démarche inductive, sans hypothèse préalable. D'autres ambivalences sont également expliquées : l'analyse peut être dirigée par l'outil ou dirigée par l'utilisateur ; l'utilisateur peut préférer une analyse de linguistique qualitative ou une analyse de linguistique quantitative.

Le premier chapitre présente des définitions du domaine de l'exploration textuelle et introduit quelques notions telles la recherche de mots dans un texte (simple ou à l'aide d'expressions rationnelles), l'annotation de corpus (permettant l'élaboration de filtres), ou encore la recherche dans une base de données à l'aide de requêtes. Il s'agit de visualiser le corpus et d'être en mesure de l'interroger ou d'élaborer des statistiques.

Le chapitre 2 est consacré aux annotations. Les auteurs en donnent une définition, expliquent que leur pose peut être manuelle ou automatique. Les difficultés, notamment dues aux différents formats de fichiers ainsi que le respect des recommandations et standards sont évoqués. Des cas d'annotations d'une grande diversité sont présentés, telles la correction d'erreurs (langage SMS, textes médiévaux), la transcription de l'oral, l'anonymisation, l'annotation de la structure du texte ou encore la pose d'annotations enrichissant le texte (syntaxe, entités nommées, références). Différents modèles d'annotations sont détaillés. La nécessité

de phases d'expérimentation lors de la phase d'annotation est argumentée. Le chapitre aborde ensuite l'exploitation des annotations à l'aide de statistiques textuelles, des segments répétés et des cooccurrences.

Le chapitre 3 intitulé « exploration de la structure d'un corpus » présente différentes approches pour explorer le contenu d'un corpus (brut ou annoté) en faisant ressortir les attractions et oppositions qu'il recèle *via* une analyse statistique (comme l'analyse factorielle des correspondances ou l'analyse en composantes principales), et la construction d'un tableau de données. Trois niveaux de données sont distingués : les mots-formes, les lemmes et les parties du discours. Les limites des approches statistiques sont expliquées (absence de traitement des hapax, par exemple) et des notions plus élaborées sont abordées comme le test du *bootstrap* ou l'effet Guttman.

Le chapitre 4 traite de l'exploration d'une hypothèse élaborée par le chercheur en se fondant sur un corpus. Deux grandes démarches sont distinguées. La première est fondée sur l'examen d'une hypothèse au regard d'une structuration du corpus ; c'est l'occasion de rappeler quelques statistiques élaborées à partir d'un tableau de contingence : test de l'écart réduit, du Khi 2, etc. La seconde est focalisée sur l'examen d'une unité linguistique spécifique en regard de la structuration du corpus (concordances, segments répétés, cooccurrences). Les limites des mesures sont expliquées.

Les auteurs ont pris soin de définir et d'expliquer les termes techniques du domaine (repérés en gras), ce qui rend cet ouvrage très utile aux personnes qui aborderaient l'exploration de corpus. De plus, l'équivalent anglais de chaque terme est signalé afin de faciliter la compréhension de l'abondante bibliographie anglophone du domaine. Le texte est bien écrit, parsemé d'exemples, certains étant développés dans des encadrés, ce qui le rend très abordable. L'ouvrage aborde également des aspects très spécialisés de l'exploration de corpus (comme les « structures de traits récursives »). Il est donc à la fois utile aux novices du domaine et aux praticiens confirmés.

Une liste d'outils d'exploration de corpus mentionnés dans l'ouvrage figure en annexe, chaque outil y est sommairement décrit. Toutefois, cette liste est incomplète, ainsi R , SATO et Alceste, bien que présents dans le cours des pages, n'y figurent pas. De nombreuses fonctionnalités, présentées dans l'ouvrage, n'apparaissent pourtant pas dans l'index ou le répertoire des outils. On pourrait regretter également l'absence d'une grille d'analyse croisant les outils et les fonctionnalités afin d'en avoir une vision synthétique. Cette absence est probablement due au foisonnement d'outils. D'ailleurs, les auteurs signalent les difficultés qu'entraînent la diversité des formats manipulés par les outils et le besoin d'une interopérabilité ou d'un outil polyvalent. Ce souhait pourra inspirer la communauté TAL œuvrant dans le domaine des outils d'exploration de corpus.

Horacio SAGGION. Automatic Text Simplification. *Morgan & Claypool publishers*. 2017. 121 pages. ISBN 9-781-62705-968-1.

Lu par **Yannis HARALAMBOUS**

IMT – UMR CNRS 6285 Lab-STICC

Cet ouvrage montre de manière très convaincante que la simplification, en tant que transformation d'un texte en un autre, est une opération très difficile. Diverses approches sont présentées, et même celles qui ne prétendent fournir que des résultats très partiels sont loin d'être efficaces. On peut citer deux principales difficultés : (a) dans la mesure du possible, le sens du texte doit rester inchangé, (b) on doit être capable de mesurer la difficulté du vocabulaire et de la syntaxe utilisés et de les comparer à ceux du texte original. Le point (b) est d'autant plus difficile que l'on a du mal à définir ce qui peut être la difficulté d'un mot ou d'une structure syntaxique.

L'auteur, Horacio Saggion, professeur à l'université Pompeu Fabra de Barcelone, a, depuis sa thèse à l'Université de Montréal en 2000, travaillé sur le résumé automatique. Il s'est ensuite spécialisé dans la simplification de textes espagnols, en particulier à destination de personnes souffrant de dyslexie, dans le cadre de projets de recherche favorisant l'intégration de personnes avec des déficiences intellectuelles.

L'ouvrage est divisé en huit chapitres. Après une introduction générale à la problématique de la simplification de texte (chapitre 1) et un historique des travaux sur la lisibilité de textes (chapitre 2) arrivent les deux chapitres les plus importants, à savoir la simplification lexicale (chapitre 3) et la simplification syntaxique (chapitre 4). Le cinquième chapitre concerne la possibilité de la simplification en tant qu'opération d'apprentissage artificiel. Enfin, les trois derniers chapitres énumèrent des systèmes de simplification, des applications de la simplification, des ressources existantes, ainsi que des méthodes d'évaluation. Le lecteur animé par la curiosité scientifique trouvera son bonheur dans les chapitres 3, 4 et 5. Celui cherchant des solutions concrètes et implémentables (voire même implémentées) pourra consulter avec profit les chapitres 6 à 8.

Chapitres 1 et 2 : introduction et la question de la lisibilité

La simplification peut cibler trois catégories principales de lecteurs : (a) les personnes souffrant de dyslexie ou de déficiences intellectuelles, (b) les personnes avec un quotient intellectuel plutôt bas, (c) les apprenants d'une langue donnée. Il y a eu des travaux dans ce sens en anglais, portugais du Brésil, japonais, français, italien, basque et espagnol. D'autre part, il existe en ligne des journaux, ou magazines, simplifiés en suédois, norvégien, français de Belgique[1], flamand, danois, italien, finnois et espagnol. Enfin, il existe une version de Wikipédia en anglais

1 Le journal *L'Essentiel*, dont la version papier a été fondée à Charleroi en 1990 par la FUNOC, un centre de formation de jour pour adultes en difficulté d'insertion socioprofessionnelle.

simplifié, qui a – comme on le verra – beaucoup servi en tant que ressource linguistique.

La *lisibilité*, introduite ici puisqu'il s'agit d'évaluer la difficulté du texte produit, est le serpent de mer de la linguistique. En effet, elle a intéressé les pédagogues et les linguistes depuis le début du XX^{e} siècle, sans jamais donner de résultats satisfaisants. Déjà dans les années 80, il y a eu des centaines de formules de lisibilité dont le but est de caractériser le niveau éducatif requis pour lire un texte donné. L'application première de ces formules était l'affectation de textes dans les manuels scolaires de différents niveaux, ainsi que l'évaluation de la difficulté des textes donnés en examen. L'auteur donne une description très sommaire du domaine (le chapitre n'occupe en tout et pour tout que treize pages), mais avec une bibliographie bien fournie et des conseils de lecture pertinents.

Chapitre 3 : simplification lexicale

On entre ici dans le vif du sujet. La méthode classique consisterait à (1) détecter les mots difficiles, (2) les désambiguïser et en trouver des synonymes, (3) choisir le synonyme le plus simple. Or, le sens et donc aussi la difficulté dépendent du contexte. Pour cette raison, la plupart des travaux se sont plutôt appuyés sur l'*alignement* de corpus, et en particulier sur l'alignement entre la Wikipédia anglais standard et celui en anglais simplifié. Ainsi, Yatskar utilise l'historique de modification de la Wikipedia simplifié pour former une base de synonymes simplificateurs (selon l'hypothèse tout à fait plausible qu'un contributeur à la Wikipedia simplifié remplacera toujours un mot par un synonyme *plus simple*). Biran utilise plutôt des vecteurs de contexte pour détecter les (quasi-)synonymes et définit la complexité d'un mot comme étant le ratio de sa fréquence dans la Wikipedia standard divisé par sa fréquence dans la Wikipedia simplifié. D'autres, enfin, utilisent des méthodes d'apprentissage profond en ce qui concerne la synonymie, et des corpus spécifiques pour l'évaluation de la difficulté, comme le corpus LexMTurk qui contient cinq cents phrases anglaises tirées de la Wikipedia contenant chacune un mot marqué pour lequel cinquante utilisateurs d'*Amazon Mechanical TUrk* ont proposé, indépendamment les uns des autres, des versions simplifiées. Toutes ces méthodes ont donné des résultats plutôt moyens, il y a juste un domaine où la simplification semble bien se passer, c'est le domaine des expressions numériques : on réussit assez bien à remplacer des expressions comme « un quart » ou « 57 % » par « ¼ » respectivement « plus de la moitié ». Mais il reste toujours des pièges lorsque, par exemple, il faut tenir compte d'un seuil de précision donné pour comparer des valeurs. L'auteur donne l'exemple de la phrase « *l'inflation au Royaume-Uni est passée de 1,2 % en septembre à 1,3 % en octobre* », simplifiée par « *l'inflation au Royaume-Uni est passée d'environ 1 % en septembre à environ 1 % en octobre* » qui pose un léger problème d'ordre pragmatique...

Chapitre 4 : simplification syntaxique

L'approche la plus classique consiste à découper les phrases longues, avec propositions subordonnées, en phrases courtes, si possible du type sujet verbe complément. Cela peut être effectué en appliquant des règles de transformation aux

arbres syntaxiques. Or, il peut y avoir des problèmes de cohésion référentielle. Voici un exemple caractéristique, tiré de l'ouvrage : *Mr. Anthony, who runs an employment agency, decries program trading, but he isn't sure it should be strictly regulated*. On y trouve donc une subordonnée relative, suivie de la proposition principale, suivie d'une deuxième proposition reliée par une conjonction de coordination. Les règles vont tout naturellement découper ces trois propositions en trois phrases, dans l'ordre (1) proposition principale, (2) subordonnée relative, (3) deuxième proposition : *Mr. Anthony decries program trading. Mr Anthony runs an employment agency. But he isn't sure it should be strictly regulated*. La permutation fait que le référent canonique du *it* de la troisième phrase n'est plus *program trading*, mais *employment agency*. Il faut donc passer par une analyse du discours, ne serait-ce que pour obtenir l'ordre des phrases découpées.

Une autre approche, qui rejoint le résumé automatique, consiste à extraire du texte les événements clés en évitant les informations de moindre importance. La technique est encore une fois fondée sur des règles de transformation.

Chapitre 5 : la simplification en tant qu'apprentissage artificiel

Ce chapitre traite de la possibilité d'appliquer des méthodes de *machine learning* sur les corpus, entre-temps assez volumineux, du Wikipedia anglais standard (plus de 5,5 millions d'articles) et du Wikipedia anglais simplifié (environ 131 milliers d'articles). Il s'agit d'appliquer des méthodes de traduction automatique, en considérant la langue simplifiée comme une langue différente de la langue de départ. Au niveau des arbres syntaxiques par constituants, les principales opérations envisagées sont la scission, la suppression, le réordonnancement, et la substitution. On calcule un modèle probabiliste fondé sur ces opérations, et on apprend à les appliquer à de nouveaux textes. Dans des travaux plus récents, on trouve également une composante sémantique, et l'utilisation de la théorie de représentation du discours de Kamp qui fournit un nouveau graphe permettant à son tour des calculs de probabilités.

Chapitres 6 à 8 : les outils et les ressources

La lecture de ces chapitres est passionnante puisque l'on y découvre un fourmillement d'idées, mais aussi une montagne de difficultés et de problèmes à résoudre. La dernière partie du chapitre 8 est consacrée à l'évaluation des systèmes de simplification. Bien évidemment, l'approche classique est de passer par des juges humains, mais certains systèmes utilisent aussi des mesures d'évaluation empruntées à la traduction automatique comme BLEU, TERp et ROUGE.

Conclusion

Cet ouvrage est une mine d'informations. Il est agréable à lire et donne un très grand nombre de pointeurs vers des publications et des ressources. C'est une synthèse claire, complète et bien argumentée du domaine par un de ses spécialistes notoires.

Le seul point négatif est d'ordre psychologique : toutes les voies empruntées dans ce livre tombent tôt ou tard sur des obstacles, ce qui engendre une certaine

frustration. On a l'impression que toutes les difficultés du TAL se sont réunies dans ce domaine et que le chercheur courageux qui veut y travailler aura fatalement à se battre contre vents et marées. Comme si cela ne suffisait pas, dans la dernière page de l'ouvrage, l'auteur souligne le fait qu'il n'y a pas une *seule* simplification, et que selon les besoins de la population ciblée, *la simplification de l'un peut aggraver la situation pour l'autre* : entre dyslexiques, apprenants d'une langue, et déficients intellectuels, il y a, du moins partiellement, incompatibilité des besoins et donc aussi des solutions.

Nous espérons que cette note de lecture attisera la curiosité du lecteur intéressé par le domaine des transformations textuelles (résumé, traduction, paraphrase, simplification) et l'incitera à la lecture de cet ouvrage, qu'il ne regrettera pas !

Émilie NÉE. Méthodes et outils informatiques pour l'analyse des discours. *Presses universitaires de Rennes*. 2017. 248 pages. ISBN 978-2-75355-499-3.

Lu par **Nadia MAKOUAR**

INALCO

Le présent ouvrage, dédié à l'analyse des données textuelles, se propose d'aborder des méthodes pratiques sous l'angle de l'analyse du discours. Les auteurs de l'ouvrage, coordonné par Émilie Née, sont tous spécialistes de l'analyse outillée du discours. Ce manuel comporte six chapitres, ponctués par des encadrés détaillant au lecteur des informations de type documentaire, bibliographique et notionnel, ainsi que des résumés de recherches menées sur des données textuelles. Un index en fin d'ouvrage reprend les notions et les termes techniques.

L'ouvrage dédié à l'analyse des données textuelles ancre d'emblée son positionnement théorique du point de vue de l'analyse du discours et développe les choix méthodologiques et pratiques qui en découlent.

Dans l'introduction les auteurs présentent de façon concise et claire les prémices de l'analyse des données textuelles. Ceci est l'occasion pour le lecteur de découvrir ou de se rappeler l'évolution des pratiques de l'analyse outillée en statistique linguistique et lexicale. Les auteurs expliquent comment la lexicométrie a progressivement laissé place à la textométrie. La mise en commun des connaissances qualitatives et quantitatives a donc permis d'étendre l'observable de l'unité lexicale au texte. Ces précisions permettent de mettre en évidence les points de divergences entre l'analyse du discours et la sémantique interprétative, mais aussi de montrer que certains travaux se trouvent à l'intersection de ces deux théories. Pour dépasser ces clivages théoriques, la communauté regroupe ces disciplines et ces courants sous l'appellation « Analyse des données textuelles » (ADT) ; l'ADT se distingue de la linguistique de corpus et du traitement automatique des langues.

L'originalité de l'ouvrage est qu'il met en regard les outils et leurs méthodes en lien avec des problématiques concrètes en analyse du discours. Tout le long du manuel, les auteurs insistent sur la nécessaire réflexion à porter sur les données

avant, pendant et après leur regroupement en corpus. Il s'agit d'une démarche itérative où les hypothèses sont constamment mises à l'épreuve.

Le chapitre 1 introduit la question du décompte des unités textuelles et les méthodes implémentées dans chacun des logiciels étudiés. Par exemple, pour un même texte, les auteurs montrent que Word, Notepad et la commande CAT sous Unix intègrent des paramètres de comptage et donc de découpage et d'identification des formes graphiques différentes. Ce qui a une incidence sur l'analyse des données du chercheur. Ils proposent également l'exemple des nuages de mots et la nécessité de prendre en compte l'ensemble des éléments du texte (notamment les mots-outils) pour mieux saisir et interpréter les données soumises au logiciel. Dans cette même perspective de sensibilisation aux données étudiées, les auteurs ont recours à Ngram Viewer (conçu par Google) qui permet d'observer l'évolution chronologique d'un mot ou d'une séquence de mots sur la base de livres numérisés par Google Livres en français. Malgré l'obtention d'un graphique, il est difficile d'en faire une quelconque interprétation en raison de l'inaccessibilité des ressources analysées. Cet exemple illustre parfaitement la problématique de la connaissance du corpus, nécessaire à une analyse objective.

C'est dans cette perspective que les auteurs proposent dans le chapitre 2 d'éclairer le lecteur sur les principes de base liés à la constitution et à la structuration du corpus. Cette partie insiste sur la réflexion à porter sur le corpus à constituer ou en voie de constitution. Ainsi, les données se construisent à partir d'hypothèses de travail. À travers ces différents principes de constitution et de structuration, les auteurs fournissent des méthodes clés que le chercheur pourrait adapter à ses propres données. Les méthodes et les stratégies d'analyse y sont explicitées. Les notions de « hors corpus » ou de « sous-corpus à géométrie variable » éclairent sur l'éventail des méthodes applicables. Les auteurs illustrent également avec quelques recherches que la diversité des approches du corpus est possible à condition de bien définir leur « statut » durant la phase d'analyse. Un rappel nécessaire est fait sur le va-et-vient permanent entre l'aspect qualitatif et le retour au texte. Les auteurs proposent d'illustrer la démarche d'analyse en fonction de plusieurs types de structuration de corpus : par genre, par sources énonciatives, par sphères d'activités et par moments discursifs. Même si un seul critère peut présider à la constitution du corpus, les spécialistes rappellent néanmoins le nécessaire croisement des critères. La question des données issues du Web y est aussi abordée ainsi que la dimension « techno-langagière » qui interroge sur le rôle joué par le support (Twitter par exemple) dans l'interprétation des données.

Le chapitre 3 propose une mise en pratique des principes explicités dans le chapitre qui le précède. Ce troisième volet de l'ouvrage s'apparente à un tutoriel où le lecteur est guidé étape par étape dans la démarche d'analyse qui commence dès l'élaboration des premières hypothèses. Ces étapes articulent les données à étudier et l'outillage convoqué pour répondre aux questionnements sur l'objet d'étude. Trois scénarios de constitution de corpus sont proposés afin d'illustrer la démarche : à partir 1) d'un corpus médiatique construit autour d'une forme langagière, 2) d'un autre corpus sociopolitique construit autour d'un thème et enfin, 3) d'un corpus politique construit autour d'un genre. Cet éventail permet alors au lecteur de prendre

connaissance des principes méthodologiques et pratiques et des pièges qui lui faut éviter afin de ne pas fausser son analyse. Le chapitre explique minutieusement le parcours de la constitution et de l'analyse en termes de délimitation, de codage, de formatage, de structuration et de balisage, et ce, en fonction du logiciel utilisé. C'est pourquoi les auteurs insistent sur la connaissance du logiciel que le lecteur souhaitera utiliser avant d'y intégrer les données. Cette partie nous éclaire aussi sur l'évolution de l'ADT et ce qu'elle a permis en termes de disponibilité et d'accessibilité des corpus « réservoirs » (comme ESLO) ou « partagés » (comme Textopol), utiles pour tout chercheur.

Le chapitre 4 qui précise la problématique des formes graphiques évoquée dans le premier chapitre, nous éclaire précisément sur le comptage des unités. Il existe en effet plusieurs façons d'aborder les données textuelles : à partir d'une forme graphique, d'un lemme, ou d'une catégorie morphosyntaxique, notamment. Plusieurs illustrations, avec des copies d'écran, montrent comment certains logiciels sont capables de traiter ces types d'unités. Les principes et les usages des segments répétés et des cooccurrences en analyse des données textuelles sont également abordés et approfondis dans ce chapitre.

Le chapitre 5 offre un panorama sur la typologie des logiciels disponibles pour l'ADT. Il introduit tout d'abord quelques repères historiques et épistémologiques sur l'apparition et le développement de différents outils d'analyse, et ce, en fonction des préoccupations et des observations des chercheurs qui les ont conçus. Les auteurs donnent l'exemple de Max Reinert qui, en cherchant à étudier des données en psychanalyse a développé le logiciel Alceste lui permettant de faire émerger les thématiques dominantes dans un texte. À partir de corpus accessibles et disponibles en ligne, les auteurs proposent d'illustrer les différentes fonctionnalités offertes principalement par les outils proposant une approche structurante, d'une part (Alceste, IRaMuTeQ), et ceux contrastifs et longitudinaux, d'autre part (Lexico, TXM, Le Trameur, etc.). Les nombreuses illustrations détaillées et expliquées précisent de façon complète les différentes approches possibles des données *via* l'utilisation de ces logiciels, toujours en fonction des hypothèses formulées et mises à l'épreuve.

Le sixième et dernier chapitre éclaire et précise au lecteur l'articulation entre les questionnements méthodologiques et le traitement des données avec les outils d'analyse. Les auteurs mettent l'accent sur les problématiques herméneutiques. Ils y abordent la question de l'approche thématique d'un corpus et proposent des points d'entrées et plusieurs méthodes quantitatives. En maintenant l'attention du lecteur sur la question de l'interprétation des données, les auteurs montrent, par exemple, les implications de l'utilisation d'une approche inductive et déductive dans le cadre d'une recherche thématique. Les recherches qui illustrent ces démarches éclairent le lecteur sur les difficultés qu'il peut lui-même rencontrer et donc, contourner. Ces analyses montrent aussi quel type de cheminement interprétatif pourrait compléter la recherche de l'analyste et ainsi enrichir ses questionnements.

En fin d'ouvrage, les auteurs consacrent quelques pages sous forme de « fiches pratiques » essentielles à la manipulation et au traitement automatique des données.

De nombreuses copies d'écran permettent au lecteur de suivre pas à pas les commandes proposées pour le traitement des données. Une dernière fiche illustrée par des schémas et des copies d'écran propose d'approfondir la notion d'analyse factorielle des correspondances ; fonctionnalité que l'on retrouve dans plusieurs logiciels d'analyse textuelle. Ces éléments sont complétés par une bibliographie et une sitographie dédiée aux notions détaillées dans ces fiches.

Cet ouvrage didactique et accessible constitue un apport scientifique et méthodologique précieux pour quiconque souhaiterait mener une recherche en analyse des textes. Il représente une source considérable d'informations théoriques et pratiques sur les différents types d'approches en ADT.

Jean-Marc Leblanc. Analyses lexicométriques des vœux présidentiels. *Wiley-Iste*. 2017. 386 pages. ISBN 978-1-78405-210-2.

Lu par **Daniel YAO**

Université Jean L. Guédé de Daloa (Côte d'Ivoire)

L'ouvrage de Jean-Marc Leblanc traite des avantages liés à l'utilisation des outils de traitement de données textuelles selon une approche lexicométrique. Il présente au niveau formel, une subdivision en six chapitres encadrés en amont, par une partie introductive et en aval par une conclusion. L'ouvrage ambitionne de présenter un traitement transversal des textes en mobilisant des expérimentations plurielles grâce à divers logiciels de traitement textuel.

L'introduction pose les grands principes qui orientent le traitement du matériau analysé en l'occurrence, les vœux présidentiels des présidents de la République française depuis Charles de Gaulle jusqu'à François Hollande. Elle insiste aussi sur les balises tant théoriques que méthodologiques à observer afin de conduire une analyse lexicométrique exploitant les multiples fonctionnalités offertes par la pluralité des logiciels de traitement textuel. Cette étape liminaire précise enfin, la nécessité de revenir toujours au texte, à la suite des manipulations statistiques, car pour reprendre l'auteur, la lexicométrie ne saurait être un raccourci méthodologique pour des analyses clé en main.

Le chapitre 1 explicite la nature du corpus à analyser (l'ensemble des discours liés aux vœux présidentiels), une brève approche historique de la lexicométrie en tant que discipline et les travaux princeps y afférents. Les premiers travaux fondateurs avaient ainsi pour ambition d'examiner la ventilation du vocabulaire, la fréquence des occurrences, d'opérer des retours réguliers au texte et de mesurer son homogénéité stylistique. Cette partie décrit et compare également les différents logiciels, selon qu'ils sont contrastifs ou longitudinaux (Lexico 3, Hyperbase, WebLex, etc.), structurants (Alceste, IRaMuTeQ, etc.) ou catégorisateurs et évaluateurs sémantiques (Tropes, Cordial, etc.), même si la tendance générale actuelle des concepteurs évolue vers leur interchangeabilité.

Le chapitre 2 opère, de manière factuelle, le traitement lexicométrique des quarante-trois messages des septennats relatifs aux vœux présidentiels. Il procède aux premières analyses des vœux présidentiels des locuteurs selon une approche diachronique et comparative. J.-M. Leblanc mobilise à ce niveau, diverses techniques telles que les AFC et des mesures de proximité sur le corpus comme la distance sur V et sur N. Ces éléments soulignent une originalité gaullienne au sein de l'homogénéité des locuteurs, et spécifient la rupture et la continuité dans lesquelles s'inscrit François Mitterrand par rapport à Charles de Gaulle. Le logiciel Lexico 3 convoqué, révèle un contraste entre les discours des septennats et ceux des quinquennats avec Jacques Chirac qui affiche une indistinction, car il se démarque peu de ses prédécesseurs. Valéry Giscard d'Estaing est proche de Georges Pompidou, avec qui il partage la même connexion du vocabulaire, tandis que François Hollande se rapproche, sur la base de son premier discours, de Nicolas Sarkozy. Ce dernier partage des liens de proximité avec Chirac sur la distance lexicale au regard des expérimentations issues des logiciels TextObserver et Hyperbase.

Dans le chapitre 3, l'auteur procède à l'étude des catégories grammaticales, des modes et des temps lexicaux liés au corpus. Il examine aussi, par le biais des logiciels Lexico, Cordial et Hyperbase, les aspects morphosyntaxiques, et les emplois des marques personnelles. Les stratégies discursives indiquent, par exemple, que Chirac privilégie les constructions du type *il faut + infinitif*, les emplois déontiques, volitifs et le présent de l'indicatif. Chez de Gaulle, le temps présent est sous-employé, les volitifs sont également présents avec une absence significative des pronoms *je, nous* et *vous* . Giscard d'Estaing, quant à lui, privilégie le *je*, le futur de l'indicatif, les emplois explicatifs dans une approche didactique et répétitive comme *cela veut dire*. Chez Pompidou, les modes subjonctif et impératif sont valorisés en des accents gaulliens sur un ton argumentatif et un équilibre entre le *je* et le *vous*. Mitterrand, tout comme Chirac, favorise la simplicité et la connivence, la personnalisation du discours où saillit l'incarnation de la fonction. Les thématiques liées à l'Europe, à la sécurité et au dialogue » sont de même prégnants. Au final, cette partie 3 expose des stratégies discursives de légitimation, de justification, d'appel à l'unité nationale avec un vocabulaire stable ou conjoncturel au sein de ruptures syntaxiques, ainsi que des profils lexico-énonciatifs contrastés chez les auteurs.

L'auteur étudie de manière spécifique, dans le chapitre 4, la notion d'*ethos* présidentiel qui caractérise l'implication personnelle du locuteur dans le discours. À travers divers outils (Lexico 3, Hyperbase, WebLex), l'approche poly-cooccurrentielle indique que le « je » présidentiel est fortement ancré dans le rituel et le métadiscursif. Le « nous » des messages est de type argumentatif accompagné des verbes d'action. Les *ethos* varient selon les présidents de la République avec une forte tendance à l'empathie chez Chirac et Pompidou tandis qu'elle est absente chez de Gaulle. Le lexicogramme récursif confirme le caractère argumentatif chez Mitterrand qui lui permet de renforcer sa légitimité. Au final, la mobilisation du logiciel Alceste, *via* les lemmatisations et les tris croisés, donne des distributions

statistiques et linguistiques faisant de Giscard, le plus grand contributeur des énoncés du « je » avec une forte représentation du rituel.

Dans le chapitre 5, en mobilisant l'outil Alceste, J.-M. Leblanc examine les « mondes lexicaux » pour obtenir des lexèmes et extraire *in fine*, les thématiques dominantes. L'auteur procède ensuite à une analyse récursive pour approfondir la compréhension des composantes du rituel *via* la structure des phrases. Il opère enfin des analyses comparatives sur la base des partitions locuteurs. Les différentes expérimentations indiquent, de prime abord, avec le logiciel Tropes que, pour de Gaulle et Mitterrand, le thème le plus représenté est « nation » avec une place importante accordée à « l'Europe » chez le second. À l'inverse, Giscard et Pompidou n'évoquent nullement ces items, leur préférant les énoncés « Français et France ». L'outil Alceste, dont l'algorithme, fondé sur la classification descendante le distingue des logiciels classiques, fournit les cinq classes sémantico-thématiques du corpus : la classe 5, la plus massive dominée par Giscard, est relative au « rituel » ; la classe 1 est liée à « la politique internationale » où de Gaulle et Mitterrand sont les plus expressifs ; la classe 4 est inhérente aux « valeurs républicaines, démocratiques, un vocabulaire incitatif et volontaire » où de Gaulle est le plus contributif ; la classe 3 relative à la « politique intérieure, économique et sociale » est le lieu commun de tous les locuteurs ; la classe 2 établit les « énoncés constatifs et bilans » mobilisés surtout par les présidents de la République du septennat et Chirac. J.-M. Leblanc effectue grâce à l'indice du Khi 2, des retours réguliers au texte (concordance), après les résultats issus de Lexico 3. Des analyses récursives sont conduites sur les sous-corpus de chaque locuteur afin d'affiner les formes significatives qui leur sont associées.

Le chapitre 6 analyse et compare les deux derniers présidents : *Sarkozy, Hollande, questions de style ?* Dans cette ultime section, l'auteur se propose de dégager les principaux profils langagiers des deux locuteurs identifiés par le biais des instruments tels que l'AFC, les classes sémantiques, les histogrammes et des recours itératifs aux textes (concordances). Il faut noter ainsi que les messages de Sarkozy deviennent chronologiquement centraux sur l'axe factoriel. Les spécificités lexicales insistent sur la permanence du rituel, indépendamment de la transformation des styles des locuteurs : de Gaulle (Algérie, coopération) ; Pompidou (nier, situation, Français) ; Giscard (vœux, bonheur, Français) ; Mitterrand (droit, Europe, désarmement) ; Chirac (avenir, solidarité, emploi) ; Sarkozy (crise, urgence, avenir) ; Hollande (vœux, compétitivité, décision). Les classes thématico-sémantiques sous Alceste, expriment avec l'analyse récursive et l'indice Khi 2, l'individuation des messages des présidents de la République. Leur vocabulaire et leurs *ethos* respectifs impriment une image spécifique à chaque discours, quoique l'influence de l'événementiel sur le rituel ne soit pas insignifiante. Le positionnement énonciatif « je, nous, vous » semble enfin déterminant dans les rapprochements perceptibles sur l'analyse factorielle entre les locuteurs. Il souligne une prégnance de l'empathie chez Sarkozy dans le rituel, caractéristique qui n'est pas étrangère chez Hollande non plus.

En conclusion, l'ouvrage de J.-M. Leblanc s'inscrit dans une double perspective : à la fois didactique, car il nous initie aux bases fondamentales de la

lexicométrie, et médium de comparaison des outils de traitement textuel. Il a recours à des cas pratiques tant dans la manipulation que dans l'interprétation des résultats tout en renvoyant le lecteur à des extensions numériques pour approfondissement sur un site Internet. Le style de rédaction de J.-M. Leblanc est accessible, nimbé d'humour et les références bibliographiques sont actuelles, riches et variées. Il s'agit donc d'une contribution pertinente dans le champ du TAL.

Résumés de thèses

Rubrique préparée par Sylvain Pogodalla

Université de Lorraine, CNRS, Inria, LORIA, F-54000 Nancy, France
sylvain.pogodalla@inria.fr

Marianne DJEMAA : marianne.djemaa@gmail.com

Titre : Stratégie domaine par domaine pour la création d'un FrameNet du français : annotations en corpus de cadres et rôles sémantiques

Mots-clés : Traitement automatique de la langue (TAL), linguistique, FrameNet, French FrameNet, sémantique lexicale, annotation sémantique, rôles sémantiques, corpus, français.

Title: *Domain by Domain Strategy for Creating a French FrameNet: Corpus Annotations of Semantic Frames and Roles*

Keywords: *Natural language processing (NLP), linguistics, FrameNet, French FrameNet, lexical semantics, semantic annotation, semantic roles, corpus, French.*

Thèse de doctorat en Linguistique théorique, descriptive et automatique, École doctorale de Sciences du Langage, Laboratoire de Linguistique Formelle (CNRS et Université Paris Diderot – Paris 7), Université Sorbonne Paris Cité, sous la direction de Marie (MC, Université Paris Diderot – Paris 7). Thèse soutenue le 14/06/2017.

Jury : Mme Marie (MC, Université Paris Diderot – Paris 7, directrice), Mme Laurence Danlos (Pr, Université Paris Diderot – Paris 7, présidente), M. Sylvain Kahane (Pr, Université Paris Ouest – Paris 10, rapporteur), Mme Marie-Claude L'Homme (Pr, Université de Montréal, Canada, rapporteur), M. Alexis Nasr (Pr, Aix-Marseille Université, examinateur).

Résumé : *Dans cette thèse, nous décrivons la création du French FrameNet (FFN), une ressource de type FrameNet pour le français créée à partir du FrameNet de l'anglais et de deux corpus arborés : le French Treebank et le Sequoia Treebank. La ressource séminale, le FrameNet de l'anglais, constitue un modèle d'annotation sémantique de situations prototypiques et de leurs participants. Elle propose à la fois :*

a) un ensemble structuré de situations prototypiques, appelées cadres, associées à des caractérisations sémantiques des participants impliqués (les rôles) ;
b) un lexique de déclencheurs, les lexèmes évoquant ces cadres ;
c) un ensemble d'annotations en cadres pour l'anglais.

Pour créer le FFN, nous avons suivi une approche « par domaine notionnel » : nous avons défini quatre « domaines » centrés chacun autour d'une notion (cause, communication langagière, position cognitive ou transaction commerciale), que nous avons travaillé à couvrir exhaustivement à la fois pour la définition des cadres sémantiques, la définition du lexique, et l'annotation en corpus. Cette stratégie permet de garantir une plus grande cohérence dans la structuration en cadres sémantiques, tout en abordant la polysémie au sein d'un domaine et entre les domaines. De plus, nous avons annoté les cadres de nos domaines sur du texte continu, sans sélection d'occurrences : nous préservons ainsi la distribution des caractéristiques lexicales et syntaxiques de l'évocation des cadres dans notre corpus. À l'heure actuelle, le FFN comporte 105 cadres et 873 déclencheurs distincts, qui donnent lieu à 1109 paires déclencheur-cadre distinctes, c'est-à-dire 1109 sens. Le corpus annoté compte au total 16167 annotations de cadres de nos domaines et de leurs rôles.

La thèse commence par resituer le modèle FrameNet dans un contexte théorique plus large. Nous justifions ensuite le choix de nous appuyer sur cette ressource et motivons notre méthodologie en domaines notionnels. Nous explicitons pour le FFN certaines notions définies pour le FrameNet de l'anglais que nous avons jugées trop floues pour être appliquées de manière cohérente. Nous introduisons en particulier des critères plus directement syntaxiques pour la définition du périmètre lexical d'un cadre, ainsi que pour la distinction entre rôles noyaux et non-noyaux.

Nous décrivons ensuite la création du FFN : d'abord, la délimitation de la structure de cadres utilisée pour le FFN, et la création de leur lexique. Nous présentons alors de manière approfondie le domaine notionnel des positions cognitives, qui englobe les cadres portant sur le degré de certitude d'un être doué de conscience sur une proposition. Puis, nous présentons notre méthodologie d'annotation du corpus en cadres et en rôles. À cette occasion, nous passons en revue certains phénomènes linguistiques qu'il nous a fallu traiter pour obtenir une annotation cohérente ; c'est par exemple le cas des constructions à attribut de l'objet.

Enfin, nous présentons des données quantitatives sur le FFN tel qu'il est à ce jour et sur son évaluation. Nous terminons sur des perspectives de travaux d'amélioration et d'exploitation de la ressource créée.

URL où le mémoire peut être téléchargé :
https://tel.archives-ouvertes.fr/tel-01661689

Yoann DUPONT : yoa.dupont@gmail.com

Titre : La structuration dans les entités nommées

Mots-clés : Reconnaissance des entités nommées, entités nommées structurées, apprentissage automatique, champs aléatoires conditionnels, réseaux de neurones.

Title: *Structuration in Named Entities*

Keywords: *Named entity recognition, structured named entities, machine learning, conditional random fields, neural networks.*

Thèse de doctorat en Sciences du Langage, Lattice, UMR 8094, Université Sorbonne Nouvelle – Paris 3, sous la direction de Isabelle Tellier (Pr, Université Sorbonne Nouvelle – Paris 3), Christian Lautier (directeur technique, Expert System France), Marco Dinarelli (CR, CNRS). Thèse soutenue le 23/11/2017.

Jury : Mme Isabelle Tellier (Pr, Université Sorbonne Nouvelle – Paris 3, codirectrice), M. Christian Lautier (directeur technique, Expert System France, codirecteur), M. Marco Dinarelli (CR, CNRS, codirecteur), Mme Agata Savary (MC HDR, Université François Rabelais Tours, rapporteur), M. François Yvon (Pr, Université Paris Sud, LIMSI/CNRS, rapporteur), M. Frédéric Landragin (DR, CNRS, examinateur), Mme Pascale Sébillot (Pr, IRISA/INSA de Rennes, examinatrice), M. Patrick Watrin (logisticien de recherche, Université catholique de Louvain, Belgique, examinateur).

Résumé : *La reconnaissance des entités nommées est une discipline cruciale du domaine du TAL. Elle sert à l'extraction de relations entre entités nommées, ce qui permet la construction d'une base de connaissances, le résumé automatique, etc. Nous nous intéressons ici aux phénomènes de structurations qui les entourent.*

Nous distinguons tout d'abord deux types d'éléments structurels dans une entité nommée. Les premiers sont des sous-chaînes récurrentes, que nous appellerons les affixes caractéristiques d'une entité nommée. Le second type d'éléments sont les tokens *ayant un fort pouvoir discriminant, appelés des* tokens *déclencheurs. Nous détaillerons l'algorithme que nous avons mis en place pour extraire les affixes caractéristiques, que nous comparerons à Morfessor. Nous appliquerons ensuite notre méthode pour extraire les* tokens *déclencheurs, utilisés pour l'extraction d'entités nommées du français et d'adresses postales.*

Une autre forme de structuration pour les entités nommées est de nature syntaxique, d'imbrications ou arborée. Pour identifier automatiquement cette structuration, nous proposons un type de cascade d'étiqueteurs linéaires qui n'avait jusqu'à présent jamais été utilisé pour la reconnaissance d'entités nommées. Elles généralisent les approches précédentes qui sont capables de reconnaître uniquement des entités de profondeur limitée ou qui ne peuvent pas modéliser certaines particularités des entités nommées structurées.

Tout au long de cette thèse, nous comparons deux méthodes par apprentissage automatique, à savoir les CRF et les réseaux de neurones, dont nous présenterons les avantages et inconvénients.

URL où le mémoire peut être téléchargé :
https://tel.archives-ouvertes.fr/tel-01772268

Mouna ELASTHER : elashtermouna@yahoo.com

Titre : Gestion et extension automatiques du dictionnaire relationnel multilingue de noms propres Prolexbase : Mise à jour multilingue et création d'un volume arabe via la Wikipédia

Mots-clés : Nom propre, Prolexbase, bases lexicales multilingues, notoriété, langue arabe, Wikipédia.

Title: *Automatic Management and Extension of the Multilingual Relational Dictionary of Proper Names Prolexbase: Multilingual Update and Creation of an Arabic Volume via the Wikipedia*

Keywords: *Proper noun, Prolexbase, multilingual lexical databases, notoriety, Arabic language, Wikipedia.*

Thèse de doctorat en Informatique, École doctorale Mathématiques, Informatique, Physique Théorique et Ingénierie des Systèmes, Laboratoire d'Informatique, Université François Rabelais de Tours, sous la direction de Denis Maurel (Pr, Université François Rabelais de Tours). Thèse soutenue le 04/07/2017.

Jury : Mme Béatrice Daille (Pr, Université de Nantes, présidenteet rapporteur), M. Kais Haddar (MC HDR, Université de Sfax, Tunisie, rapporteur), Mme Béatrice Markhoff (MC HDR, Université François Rabelais de Tours, examinatrice), M. Denis Maurel (Pr, Université François Rabelais de Tours, directeur).

Résumé : *Les bases de données lexicales jouent un rôle important dans plusieurs domaines du traitement automatique des langues (TAL), comme l'extraction d'information, la reconnaissance d'entités nommées et la traduction automatique des noms propres. Toutefois, elles nécessitent un développement et un enrichissement permanents par l'exploitation des ressources libres et riches en textes du web sémantique, entre autres, l'encyclopédie universelle Wikipédia, DBpedia, Geonames et Yago2.*

Le dictionnaire électronique relationnel multilingue de noms propres, Prolexbase, issu de nombreux travaux de recherche sur le TAL, comporte à ce jour dix langues, parmi lesquelles trois sont bien couvertes : le français, l'anglais et le polonais. Il a été conçu manuellement et une première tentative semi-automatique a été réalisée par le projet ProlexFeeder. Notre travail avait pour objectif d'élaborer un outil de mise à jour et d'extension automatique de ce lexique, et l'ajout de la langue arabe. Tout d'abord, une mise à jour multilingue de la base de données a été effectuée grâce à l'établisse-

ment d'un système automatique de consolidation des liens Wikipédia dans Prolexbase en nous servant du concept interlangue de Wikipédia. En conséquence, un nombre considérable de nouveaux liens Wikipédia a été ajouté dans toutes les langues constituant la base de données, et cet ajout a été précédé, le cas échéant, d'un traitement des redirections.

Un système entièrement automatique a également été mis en place qui permet de calculer, via l'encyclopédie Wikipédia, un indice de notoriété pour les entrées de Prolexbase. Cet indice dépend de la langue et participe, d'une part, à la construction d'un module de Prolexbase pour la langue arabe et, d'autre part, à la révision de la notoriété actuellement présente pour les autres langues de la base. Pour calculer la notoriété, une technique multicritères de l'aide à la décision a été utilisée : la méthode SAW incluant le calcul de l'entropie de Shannon, à partir de cinq valeurs numériques déduites de l'encyclopédie Wikipédia. Finalement, l'utilisation des liens Wikipédia a été l'instrument fondamental pour la création d'un volume arabe dans Prolexbase par un processus d'extraction de noms propres arabes depuis leurs liens Wikipédia obtenus précédemment.

URL où le mémoire peut être téléchargé :
https://hal.archives-ouvertes.fr/tel-01657366

Jérémy FERRERO : ferrero.jerem@gmail.com

Titre : Similarités textuelles sémantiques translingues : vers la détection automatique du plagiat par traduction

Mots-clés : Plagiat, détection de plagiat, détection de similarité, translingue, traduction.

Title: *Cross Lingual Semantic Textual Similarity Detection: towards Cross-Language Plagiarism Detection*

Keywords: *Cross-language, cross-lingual, plagiarism, plagiarism detection, similarity detection.*

Thèse de doctorat en Informatique, Laboratoire Informatique de Grenoble (LIG), UFR Informatique, mathématiques, mathématiques appliquées de Grenoble, Université Grenoble Alpes, sous la direction de Laurent Besacier (Pr, Université Grenoble Alpes). Thèse soutenue le 08/12/2017.

Jury : M. Laurent Besacier (Pr, Université Grenoble Alpes, directeur), M. Didier Schwab (MC, Université Grenoble Alpes, examinateur), Mme Isabelle Tellier (Pr, Université Sorbonne Nouvelle – Paris 3, présidente), M. Emmanuel Morin (Pr, Université de Nantes, rapporteur), M. Juan-Manuel Torres-Moreno (MCHDR, Université d'Avignon et des Pays de Vaucluse, rapporteur), M. Frédéric Agnès (ingénieur, Compilatio, examinateur).

Résumé : *La mise à disposition massive de documents via Internet (pages Web, entrepôts de données, documents numériques, numérisés ou retranscrits, etc.) rend de plus en plus aisée la récupération d'idées. Malheureusement, ce phénomène s'accompagne d'une augmentation des cas de plagiat.*

En effet, s'approprier du contenu, peu importe sa forme, sans le consentement de son auteur (ou de ses ayants droit) et sans citer ses sources, dans le but de le présenter comme sa propre œuvre ou création, est considéré comme plagiat. De plus, ces dernières années, l'expansion d'Internet a également facilité l'accès à des documents du monde entier (écrits dans des langues étrangères) et à des outils de traduction automatique de plus en plus performants, accentuant ainsi la progression d'un nouveau type de plagiat : le plagiat translingue. Ce plagiat implique l'emprunt d'un texte tout en le traduisant (manuellement ou automatiquement) de sa langue originale vers la langue du document dans lequel le plagiaire veut l'inclure. De nos jours, la prévention du plagiat commence à porter ses fruits, grâce notamment à des logiciels antiplagiat performants qui reposent sur des techniques de comparaison monolingue déjà bien éprouvées. Néanmoins, ces derniers ne traitent pas encore de manière efficace les cas translingues. Cette thèse est née du besoin de Compilatio, une société d'édition de l'un de ces logiciels antiplagiat, de mesurer des similarités textuelles sémantiques translingues (sous-tâche de la détection du plagiat).

*Après avoir défini le plagiat et les différents concepts abordés au cours de cette thèse, nous établissons un état de l'art des différentes approches de détection du plagiat translingue. Nous présentons également les différents corpus déjà existants pour la détection du plagiat translingue et exposons les limites qu'ils peuvent rencontrer lors d'une évaluation de méthodes de détection du plagiat translingue. Nous présentons ensuite le corpus que nous avons constitué et qui ne possède pas la plupart des limites rencontrées par les différents corpus déjà existants. Nous menons, à l'aide de ce nouveau corpus, une évaluation de plusieurs méthodes de l'état de l'art et découvrons que ces dernières se comportent différemment en fonction de certaines caractéristiques des textes sur lesquelles elles opèrent. Ensuite, nous présentons de nouvelles méthodes de mesure de similarités textuelles sémantiques translingues basées sur des représentations continues de mots (*word embeddings*). Nous proposons également une notion de pondération morphosyntaxique et fréquentielle de mots, qui peut aussi bien être utilisée au sein d'un vecteur qu'au sein d'un sac de mots, et nous montrons que son introduction dans ces nouvelles méthodes augmente leurs performances respectives. Nous testons ensuite différents systèmes de fusion et combinaison entre différentes méthodes et étudions les performances, sur notre corpus, de ces méthodes et fusions en les comparant à celles des méthodes de l'état de l'art. Nous obtenons ainsi de meilleurs résultats que l'état de l'art dans la totalité des sous-corpus étudiés. Nous terminons en présentant et discutant les résultats de ces méthodes lors de notre participation à la tâche de similarité textuelle sémantique (STS) translingue de*

la campagne d'évaluation SemEval 2017, pour laquelle nous nous sommes classés 1ers pour la sous-tâche correspondant le plus au scénario industriel de Compilatio.

URL où le mémoire peut être téléchargé :
https://tel.archives-ouvertes.fr/tel-01721390

Dhaou GHOUL : dhaou.ghoul@gmail.com

Titre : Classifications et grammaires des invariants lexicaux arabes en prévision d'un traitement informatique de cette langue. Construction d'un modèle théorique de l'arabe : la grammaire des invariants lexicaux temporels

Mots-clés : Corpus, classification, désambiguïsation, environnement syntaxique, expression régulière, langue arabe, invariants lexicaux, identification, règles linguistiques, grammaire régulière, schémas de grammaires, TAL.

Title: *Classifications and Grammars of Arab Lexical Invariants in Anticipation of an Automatic Processing of this Language. Construction of a theoretical Model of the Arabic Language: The Temporal Invariants*

Keywords: *Corpus, classification, disambiguation, syntactic environment, regular expression, Arabic language, lexical invariants, identification, linguistic rules, regular grammar, diagrams of grammars, NLP.*

Thèse de doctorat en linguistique, section informatique, Université Paris-Sorbonne, sous la direction de Amr Helmy Ibrahim (Pr émérite, Université Paris-Sorbonne et Université de Franche-Comté, Besançon). Thèse soutenue le 07/12/2016.

Jury : M. Amr Helmy Ibrahim (Pr émérite, Université Paris-Sorbonne et Université de Franche-Comté, Besançon, directeur), M. Mounir Zrigui (Pr, Université de Monastir, Tunisie, rapporteur et président), M. Mohamed Embarki (Pr, Université de Franche-Comté, Besançon, rapporteur), M. André Jaccarini (CR, CNRS, IRAA, examinateur).

Résumé : *Cette thèse porte sur la classification et le traitement des invariants lexicaux arabes qui sont des marqueurs de temporalité et d'aspect. Nous avons associé à chaque invariant des schémas de grammaire (sous forme d'automates). Dans ce travail, nous avons limité notre traitement à vingt invariants lexicaux. Notre hypothèse est construite à partir du principe selon lequel les invariants lexicaux sont situés au même niveau structural que les schèmes dans le langage quotient (squelette) de la langue arabe. Ils recèlent beaucoup d'informations et entraînent des attentes syntaxiques qui permettent de prédire la structure de la phrase. Au début de cette thèse, nous abordons la notion d'*invariant lexical *en exposant les différents niveaux d'invariance. Ensuite, nous classons les invariants étudiés dans cette thèse selon plusieurs critères. La deuxième partie de cette thèse concerne les invariants lexicaux temporels. Nous commençons par une présentation de notre méthode d'étude linguistique ainsi*

que la modélisation par schémas de grammaires associés aux invariants lexicaux temporels étudiés. Ensuite, nous abordons l'analyse proprement dite des invariants lexicaux simples (comme ḥattā*,* baʿda*) et complexes (comme* baʿdamā*,* baynamā*). Enfin, une application expérimentale, Kawâkib, a été employée pour détecter et identifier les contextes de ces invariants lexicaux. Nous montrons les points forts de ses fonctionnalités ainsi que ses lacunes. Nous proposons également une nouvelle vision de la prochaine version de Kawâkib qui peut aussi représenter une application pédagogique de l'arabe avec un recours au lexique minimal.*

URL où le mémoire peut être téléchargé :
http://www.mmsh.univ-aix.fr/program/Documents/GHOUL_Dhaou_2016_These.pdf

---

Pierre-Antoine JEAN : pierreantoine.jean@gmail.com

Titre : Gestion de l'imprécision et de l'incertitude dans un processus d'extraction de connaissances

Mots-clés : Extraction de connaissances, TAL, inférence, représentation des connaissances, incertitude.

Title: *Imprecision and Uncertainty Management in a Knowledge Extraction Process*

Keywords: *Knowledge extraction, NLP, inference, knowledge representation, uncertainty.*

Thèse de doctorat en Informatique, Laboratoire de Génie Informatique et d'Ingénierie de Production, école doctorale Information Structures Systèmes, Université de Montpellier, sous la direction de Jacky Montmain (Pr, IMT Mines Alès)et Patrice Bellot (Pr, Aix-Marseille Université, Laboratoire des Sciences de l'Information et des Systèmes, UMR 7296). Thèse soutenue le 23/11/2017.

Jury : M. Jacky Montmain (Pr, IMT Mines Alès, codirecteur), M. Patrice Bellot (Pr, Aix-Marseille Université, Laboratoire des Sciences de l'Information et des Systèmes, UMR 7296, codirecteur), Mme Catherine Berrut (Pr, Université de Grenoble, rapporteur), M. Pierre Zweigenbaum (DR, CNRS, LIMSI, Orsay, rapporteur), Mme Béatrice Daille (Pr, Université de Nantes, présidente), M. Mathieu Roche (MC, CIRAD, examinateur), M. Sébastien Harispe (MC, IMT Mines Alès, encadrant), Mme Sylvie Ranwez (Pr, IMT Mines Alès, encadrant).

Résumé : *Les concepts de découverte et d'extraction de connaissances ainsi que d'inférence sont abordés sous différents angles au sein de la littérature scientifique. En effet, de nombreux domaines s'y intéressent allant de la recherche d'information, à l'implication textuelle en passant par les modèles d'enrichissement automatique des bases de connaissances. Ces concepts suscitent de plus en plus d'intérêt à la fois dans le monde académique et industriel favorisant le développement de nouvelles méthodes.*

Cette thèse propose une approche automatisée pour l'inférence et l'évaluation de connaissances basée sur l'analyse de relations extraites automatiquement à partir de textes. L'originalité de cette approche repose sur la définition d'un cadre tenant compte (i) de l'incertitude linguistique et de sa détection dans le langage naturel, réalisée au travers d'une méthode d'apprentissage tenant compte d'une représentation vectorielle spécifique des phrases, (ii) d'une structuration des objets étudiés (p. ex. syntagmes nominaux) sous la forme d'un ordre partiel tenant compte à la fois des implications syntaxiques et d'une connaissance a priori formalisée dans un modèle de connaissances de type taxonomique (iii) d'une évaluation des relations extraites et inférées grâce à des modèles de sélection exploitant une organisation hiérarchique des relations considérées. Cette organisation hiérarchique permet de distinguer différents critères en mettant en oeuvre des règles de propagation de l'information permettant ainsi d'évaluer la croyance qu'on peut accorder à une relation en tenant compte de l'incertitude linguistique véhiculée.

Bien qu'a portée plus large, notre approche est ici illustrée et évaluée au travers de la définition d'un système de réponse à un questionnaire, généré de manière automatique, exploitant des textes issus du Web. Nous montrons notamment le gain informationnel apporté par la connaissance a priori, l'impact des modèles de sélection établis et le rôle joué par l'incertitude linguistique au sein d'une telle chaîne de traitement. Les travaux sur la détection de l'incertitude linguistique et la mise en place de la chaîne de traitement ont été validés par plusieurs publications et communications nationales et internationales. Les travaux développés sur la détection de l'incertitude et la mise en place de la chaîne de traitement sont disponibles au téléchargement à l'adresse suivante : https://github.com/PAJEAN/.

URL où le mémoire peut être téléchargé :
https://www.researchgate.net/publication/323458149_Gestion_de_l'incertitude_et_de_l'imprecision_dans_un_processus_d'extraction_de_connaissances_a_partir_des_textes

---

Rachel PANCKHURST : rachel.panckhurst@univ-montp3.fr

Titre : Entre linguistique et informatique. Des outils de traitement automatique du langage naturel écrit (TALNE) à l'analyse du discours numérique médié (DNM)

Mots-clés : Traitement automatique du langage naturel écrit (TALNE) en français, analyse de discours numériques médiés (courriels, forums, chats, SMS, WhatsApp, messageries instantanées), dispositifs novateurs en eLearning, évaluation de logiciels.

Title: *Between Linguistics and Computational Linguistics. From Written Natural Language Processing Tools to Mediated Digital Discourse Analysis*

Keywords: *Written natural language processing in French, mediated digital discourse analysis (email, forums, chats, SMS, WhatsApp, instant messaging), innovative eLearning methods, software evaluation.*

Habilitation à diriger des recherches en Informatique, Université Paris-Est, sous la direction de Panayota Kyriacopoulou (Pr, Université Paris-Est Marne-la-Vallée). Habilitation soutenue le 30/05/2017.

Jury : M. Georges Antoniadis (Pr, Université Grenoble-Alpes, examinateur), M. Cédrick Fairon (Pr, Université catholique de Louvain, Belgique, rapporteur), Mme Cvetana Krstev (Pr, Université de Belgrade, Serbie, examinatrice), Mme Panayota Kyriacopoulou (Pr, Université Paris-Est Marne-la-Vallée, directrice), M. Éric Laporte (Pr, Université Paris-Est Marne-la-Vallée, rapporteur), Mme Claudine Moïse (Pr, Université Grenoble-Alpes, examinatrice), M. Mathieu Roche (chercheur HDR, Cirad, UMR TETIS, Montpellier, examinateur), Mme Frédérique Segond (directrice du centre de R&D, Viseo, Grenoble, rapporteur).

Résumé : *Mon habilitation à diriger des recherches, intitulée « Entre linguistique et informatique. Des outils de traitement automatique du langage naturel écrit (TALNE) à l'analyse du discours numérique médié (DNM) », soutenue à la COMUE Université Paris-Est, se décline en trois volumes : volume I (synthèse), volume II (publications), volume III (curriculum vitæ)*[1]*. Ce résumé ne concerne que le volume I (synthèse).*

Depuis mon doctorat et ma nomination en tant qu'enseignante-chercheuse à l'université Paul-Valéry Montpellier 3 (en octobre 1992), mes activités d'enseignement, d'administration et de recherche s'inscrivent dans le domaine du traitement automatique du langage et des langues (TAL), et, plus précisément, du traitement automatique du langage naturel écrit (TALNE). Trois cheminements, ou volets de recherche, traversent et s'imbriquent tout au long de mes 25 années de carrière universitaire, jusqu'à présent :

1) prototypes et outils : interrogatives, verbes, gloses (1991-2003) ;

2) formation, (auto)évaluation, réseaux pédagogiques (technologies de l'information et de la communication éducatives (TICE), formation ouverte et à distance (FOAD, eLearning*)) (1996-2012) ;*

3) communication médiée par ordinateur (CMO), discours électronique médié (DEM), DNM : analyse de courriels, forums, chats, SMS (1996-2017).

Ceux-là sont explorés tout au long de ce manuscrit. La façon dont la recherche s'imprègne et s'enrichit de mes activités d'enseignement et d'administration est, je crois, cruciale. De ce fait, je présente l'ensemble de mes activités tripartites en tant qu'enseignante-chercheuse, afin que le lecteur puisse mieux entrevoir mon parcours global.

1. Les volumes II et III sont disponibles sur demande à l'adresse électronique indiquée ci-dessus.

À travers mon parcours, certes atypique — dans la mesure où je n'ai découvert l'espace francophone au quotidien qu'à partir de 18 ans —, j'espère pouvoir montrer comment j'ai contribué en recherche (mais également en pédagogie et en administration) au domaine de la linguistique informatique et, par conséquent, de quelle manière j'estime être en mesure d'animer des recherches doctorales. Dans une première section, je dessine brièvement mon parcours initial jusqu'au doctorat — afin de guider le lecteur à travers mes tout premiers pas de jeune chercheuse. La deuxième section est consacrée à l'évocation de mes activités d'enseignement et de formation et de mes responsabilités pédagogiques et administratives. Cela n'est peut-être pas habituel dans le cadre d'une habilitation, mais je m'octroie le droit de le faire, dans la mesure où je souhaite mettre en lumière leur importance pour moi et indiquer comment elles ont nourri ma réflexion en recherche. La troisième section constitue le « noyau dur » du manuscrit : la recherche. J'explique comment j'ai tissé les fils des volets et des thématiques, comment j'ai tâtonné, bifurqué, au gré des rencontres scientifiques. Puis, je montre aussi comment l'enseignement, l'administration et la recherche s'imbriquent, de manière plus approfondie. Dans la quatrième et dernière section, je mentionne les horizons et les perspectives à venir, avant de proposer une sélection globale de mes publications. Parmi les directions futures citées et les points clefs que j'estime fondamentaux à retenir dans un cadre de recherche publique, je les énumère ainsi en anglais dans ma conclusion — peut-être pourrait-on m'accuser d'être utopique ?

1) Deliver crucial research information to the general public *(fournir au grand public des informations de recherche cruciales).*

2) Request cross-disciplinary PhDs *(demander des doctorats interdisciplinaires).*

3) Demand that research results be factored into Ministerial reforms *(exiger que les résultats de la recherche soient pris en compte dans les réformes ministérielles).*

4) Help provide scientific expertise for devising real-life applications and software *(aider à fournir une expertise scientifique afin de concevoir des applications et des logiciels).*

5) Continue applied research—including PhD supervision—and help link up academic institutions with other organisations. *(poursuivre la recherche appliquée — y compris dans le cadre doctoral — et aider à établir des réseaux entre les établissements universitaires et d'autres organisations).*

URL où le mémoire peut être téléchargé :
https://hal.archives-ouvertes.fr/tel-01646172/

Marie-Sophie PAUSÉ : pauselinguist@gmail.com

Titre : Structure lexico-syntaxique des locutions du français et incidence sur leur combinatoire

Mots-clés : Locution, lexicologie, phraséologie du français, interface sémantique-syntaxe, Lexicologie Explicative et Combinatoire (LEC), Réseau Lexical du Français (RL-fr).

Title: *Impact of Lexico-Syntactic Structure of French Idioms on their Combinatory*

Keywords: *Idiom, lexicology, phraseology of the French language, semantic-syntax interface, Explanatory Combinatorial Lexicology (ECL), French Lexical Network (fr-LN).*

Thèse de doctorat en Sciences du langage, ATILF (CNRS et Université de Lorraine), Nancy, sous la direction de Alain Polguère (Pr, Université de Lorraine) et Sylvain Kahane (Pr, Université Paris Ouest – Paris 10). Thèse soutenue le 03/11/2017.

Jury : M. Alain Polguère (Pr, Université de Lorraine, codirecteur), M. Sylvain Kahane (Pr, Université Paris Ouest – Paris 10, codirecteur), M. Xavier Blanco (Pr, Universitat Autònoma de Barcelona, Espagne, rapporteur), Mme Christiane Fellbaum (Senior Research Scholar, Princeton University, États-Unis, rapporteur), Mme Éva Buchi (DR, CNRS, ATILF, Nancy,), Mme Agata Savary (MC HDR, Université François Rabelais de Tours, examinatrice).

Résumé : *En tant que syntagmes sémantiquement non compositionnels, les locutions sont des unités lexicales à part entière, qui doivent avoir leur propre entrée dans un modèle du lexique. Elles doivent donc recevoir une définition lexicographique, ainsi qu'une description de leurs caractéristiques grammaticales. De plus, en vertu de leur signifiant syntagmatique, les locutions témoignent — à des degrés divers — d'une flexibilité formelle (passivation, insertion de modificateurs, substitution de certains constituants, etc.).*

Cette thèse défend l'idée selon laquelle une description des locutions combinant à la fois l'identification des unités lexicales qui les composent et l'identification des relations de dépendance syntaxique qui unissent ces unités lexicales, permettra de prédire leurs différents emplois possibles dans la phrase. Une telle description n'est possible que dans un modèle du lexique décrivant précisément la combinatoire des lexies. Notre recherche, basée sur les principes de la Lexicologie Explicative et Combinatoire, exploite et enrichit les données du Réseau Lexical du Français (RL-fr), ressource en cours de développement à l'ATILF.

La thèse a deux principaux apports. Le premier est le développement d'un modèle de description lexico-syntaxique relativement fine des locutions du français. Le second est l'identification et l'étude de différentes variations structurales, syntaxiques et lexicales liées à la flexibilité formelle des locutions. Les variations des locutions sont mises en corrélation avec leurs structures lexico-syntaxiques, mais également avec

leurs définitions lexicographiques. Ceci nous conduit à introduire la notion de projection structurale, centrale dans le continuum de la flexibilité formelle des locutions.

La thèse est structurée en cinq chapitres, dont une introduction générale et une conclusion générale. L'introduction générale présente l'objet d'étude, les objectifs de la thèse, et la méthodologie employée. La notion de locution est présentée sous l'angle lexicographique, ce qui implique une mise en lumière des limites des modélisations lexicographiques des locutions françaises qui précèdent la modélisation préconisée dans le cadre de ce travail, à savoir celle du Réseau Lexical du français.

Le chapitre second a pour objectif, d'une part, de présenter les notions de base, et, d'autre part, de développer les caractéristiques des locutions en les positionnant par rapport aux autres classes de phrasèmes.

Le troisième chapitre présente les principes de description lexicographique des locutions que ce travail a permis d'établir dans le cadre du développement du Réseau lexical du français. Les locutions sont des unités lexicales à part entière, qui constituent des nœuds du réseau, au même titre que les lexèmes. Lexèmes et locutions appartiennent à la catégorie des lexies. La description lexicographique des locutions inclut des caractéristiques grammaticales, une définition, et des liens paradigmatiques et syntagmatiques avec d'autres nœuds du réseau. Parmi les caractéristiques grammaticales figure une structure lexico-syntaxique. Cette dernière permet d'identifier le patron syntaxique et les unités lexicales constituantes des locutions.

Le quatrième chapitre de la thèse présente le produit de la description lexico-syntaxique des locutions obtenu au terme de nos trois années de travaux — soit 498 patrons pour 2 821 locutions. Les patrons syntaxiques correspondent chacun à une structure syntaxique de surface, mais sont présentés sous format linéaire. Les positions actancielles contrôlées par certaines locutions sont prises en considération, et donnent lieu à des patrons spécifiques. La classification des patrons est opérée relativement aux types grammaticaux de locutions dénombrés (locutions verbales, locutions nominales, etc.). Le nombre de locutions associées à chaque patron est indiqué.

Reprenant la distinction opérée, au second chapitre, entre flexibilité formelle et défigement d'une locution, le chapitre cinq propose une classification puis une modélisation des variations formelles des locutions à partir d'exemples attestés. La dernière section du chapitre est consacrée à la description des variations formelles de 47 locutions verbales construites sur un patron syntaxique linéarisé du type `V Art NC`*. Les variations formelles suivantes sont étudiées : passivation, clivage, relativisation, variabilité du déterminant du constituant nominal, et attachement d'un dépendant syntaxique à un constituant autre que la tête de syntagme. Une modélisation à l'interface entre sémantique et syntaxe est proposée, sous l'angle de la projection structurale. Cette notion est introduite afin de rendre compte des correspondances entre des sémantèmes du réseau sémantique de la définition de la locution et des constituants de sa structure lexico-*

syntaxique. Les correspondances identifiées permettent d'activer certaines variations formelles.

URL où le mémoire peut être téléchargé :
http://hal.archives-ouvertes.fr/tel-01657880