AUGMENTED LEOS
FINAL REPORT

Overview of smart functionalities in drafting legislation in LEOS

February 2024

Fotios Fitsilis

George Mikros

Sotiris Leventis

Augmented LEOS
final report

Authors

Dr. Fotios Fitsilis, Hellenic Parliament
Dr. George Mikros, Hamad Bin Khalifa University
Sotiris Leventis, Hypernetica

Contact

Directorate-General for Digital Services, European Commission
1049 Bruxelles/Brussel, Belgium, Belgium
DIGIT-COMM-TEAM@ec.europa.eu
Hellenic OCR Team
5B Aretis st., 15343 Agia Paraskevi, Greece
info@hellenicOCRteam.gr

Disclaimer

Table of Contents

Executive summary

The report presents the findings of a study conducted between June 2023 and February 2024 to assess the potential of introducing smart functionalities to the Legislation Editing Open Software (LEOS) system. The goal of the study was to contribute to developing LEOS into a future-proof tool that can support the entire legislative process. It builds on a previous reference study that identified 34 potential smart functionalities grouped into nine categories.

Through interviews and questionnaires with experts from the European Commission (EC), the study refined the initial categories into seven distinct ones: verification, change tracking, linguistic support, legal assistance, automated drafting, legal practices, and policy dimension. Subsequently, it prioritised 11 smart functionalities for in-depth analysis.

The prioritised smart functionalities were investigated in terms of user experience needs, business value, relevant technology, data requirements, and performance considerations. In addition, five core technologies were identified as suitable for implementing them: Advanced Language Editing and Correction, Named Entity Recognition, Semantic Similarity, Natural Language Generation, and Information Extraction.

The study outlined common attributes around technology, data, and performance, and then analysed the distinct attributes of each prioritised functionality. The findings indicate that smart functionalities can be grouped into technology clusters, suggesting that they can be deployed using similar technologies. Furthermore, various smart functionalities may be merged, as they provide similar features. The potential technical approach will involve open source tools, where feasible, to accelerate development timelines.

A high-level roadmap has been proposed, outlining a two-year timeframe for the development of the infrastructure, regulatory framework, and a series of proofs-of-concept representing distinct technology clusters. This groundwork sets the stage for implementing an “augmented LEOS” system at the production level. It excluded the Natural Language Generation technology due to extremely rapid advancements in the Large Language Models (LLMs) field with dependencies on model selection, data infrastructure, and training.

Some of the considerations that the study identified were privacy, security, legal interoperability, training requirements, and aligning with regulations like the upcoming AI Act. Such issues need to be taken into account during the planning and the software development process. It also noted that while AI technologies evolve rapidly, the key focus areas remain valid starting points. Moreover, it revealed that most smart functionalities can be implemented without the need to resort to LLMs. The study concluded by summarising its approach and contributions toward developing an "augmented LEOS" capable of effectively supporting existing legal processes at EC level exploiting state-of-the-art technology.

1 Introduction

1.1 Study background and objective

This report is the main outcome of the EU-funded project entitled "Overview of Smart Functionalities in Drafting Legislation in LEOS" or, alternatively, "Augmented LEOS", which was implemented between June 2023 and February 2024. The project, which takes the form of a study, aims to prepare and contribute to the further progress of LEOS/EdiT into a future-proof, complete, user-assistive drafting tool that can be used throughout the complete legislative life cycle.[1] In particular, the study's outcome will contribute in the preparation of the LEOS proposal for the Digital Europe Programme 2025-2026 and the LEOS Work Plan 2024. As such, the content of this report builds upon a previous study entitled "Drafting Legislation in the era of AI and Digitisation", which was delivered in 2022.

The main focus of the study targets the business value assessment of smart functionalities and their techno-business feasibility. In this regard, it needs to be mentioned that business value and business feasibility are two separate concepts in business analysis. In essence, while business value focuses on the benefits and outcomes of a certain smart functionality, business feasibility evaluates the practicality and likelihood of successfully implementing it. Hence, the focus of the study will be placed on selected aspects of the technical feasibility of implementation and deployment of smart functionalities indicated by a group of selected European Commission (EC) experts.

While the entire original list of 34 smart functionalities will undergo assessment, particular emphasis will be placed on a subset of 11 items that were prioritised based on qualitative evaluations of a series of questionnaires. These smart functionalities will be clustered into groups of technologies, with significant information about the algorithms that may be used to achieve the desired functionality. Moreover, specific directions in their developing process and integration into LEOS will be derived.

1.2 Work plan and deliverables

The study began in June 2023 and was concluded in eight months. It consisted of four distinct tasks: project management (*Task 01*); categories of smart functionalities (*Task 02*), business value and techno-business feasibility of the implementation and deployment of categories of smart functionalities (*Task 03*), and high-level roadmap (*Task 04*). Each task was linked to a separate mandatory report. The implementation of the tasks will follow a structured process that follows the timeline shown in *Figure 1.1*.

A hybrid, decentralised approach to conducting the study was considered most efficient, with the project experts contributing from different locations. A mission to Brussels was contemplated as necessary for coordination, preliminary presentation of results and data collection purposes.

[1] LEOS: Legislation Editing Open Software; EdiT is the LEOS instance used by the European Commission (EC).

The mission took place on 25 September 2023 and its timing was carefully determined for the project team to be able to meaningfully contribute to the SEMIC 2023 conference, the annual conference on semantic interoperability organised by the European Commission that took place on 17-18 October 2023.

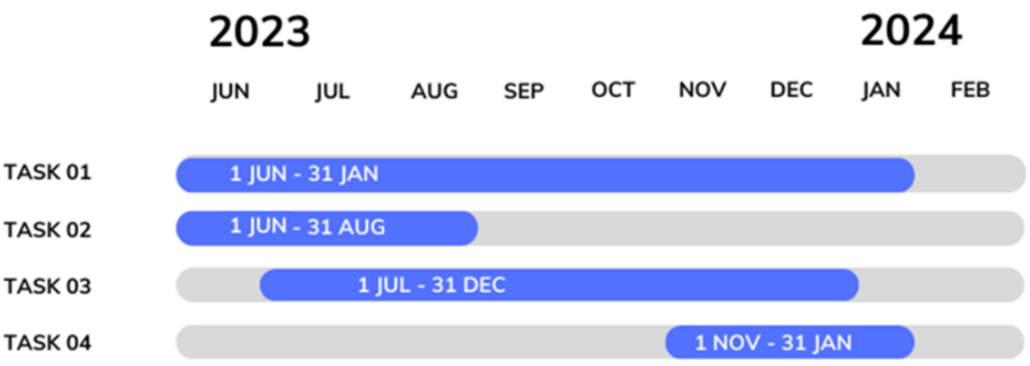


Figure 1.1. Study timeline.

Project management (Task 01) was divided into three phases: planning, executing, and closing. The general work plan was first developed based on known information and data collected during project initiation. Apart from the standard kick-off/closing meetings, frequent meetings were necessary to present the progress made and raise any issues to be tackled on the European Commission side. Overall, ten such meetings were held during the study. In addition, brief monthly reports described the activities vis-à-vis the necessary requirements as defined in the project work plan. This final study report was also part of the project management phase.

The next task, categories of smart functionalities (Task 02), included a research summary and the analysis of the categories of smart functionalities. The experts assessed the smart functionalities that had been already identified. The relevant reports and literature were also studied. Further feedback from LEOS use cases was considered. In addition, discussions and interviews with LEOS users and developers were conducted. The primary output of Task 02 included the definition of (more) coherent categories of smart functionalities for law-making/policy development by public actors, mainly those at EU level.

Task 03 constituted the core activity of the study. The experts analysed in detail both the business value and the techno-business feasibility of implementing and deploying the smart functionalities that were identified under Task 02. Analysis was conducted in three main steps. The first step documented the business value of smart functionalities. For this, relevant EC actors were consulted, e.g. via questionnaires and/or interviews. A second step concerned the technical feasibility of the determined functionalities. In a third step, the business feasibility of smart functionalities was assessed, while combining the insights from the two first steps together with related defining parameters and attributes.

The fourth and final task led to the development of a high-level roadmap (Task 04) for the implementation and deployment of the smart functionalities that were closely defined in the previous parts of the study. Efforts were invested in the roadmap to be as ripe for implementation as possible. The pragmatic approach relied on the expertise of the project team in software planning but also in implementing LEOS-based solutions. *Table 1.1* displays the main work products of the study and the timeline of their delivery.

Task	Deliverable	Description	Project month
01	D01.01	Project work plan	1
01	D01.02	Progress reports	monthly
01	D01.03	Final report	8
02	D02.01	Summary of the desk research	1
02	D02.02	Description of the categories and smart functionalities	3
03	D03.01	Business value assessment of smart functionalities	7
03	D03.02	Succinct report on techno-business feasibility of smart functionalities	7
04	D04.01	High-level roadmap	8

Table 1.1. Study tasks and deliverables.

The current report consolidates the results of all study deliverables into a single document. The report's structure and contents are presented in the next section.

1.3 Report structure and content

The current part of the study's final report presents its structure and contents in detail. Following the introductory section (*Section 1*), the current status in the field is shown (*Section 2*). For this, a thorough examination and analysis of existing information and resources was conducted. This was considered necessary to gain insights in order to identify patterns and draw first conclusions without having to resort to new data collection. In the context of the current study on Augmented LEOS, specific sources of information were tapped, for instance EC experts and related material, scientific literature and reporting, and the latest AI-based tools and services. The reviewed material was limited to a predefined set of platforms, tools and documents, which mainly consist of the following: the reference study; literature and reporting on legal informatics and Artificial Intelligence (AI); European Commission material; LEOS documentation and related material; and rest material (existing platforms and tools, including Large Language Models).

Given the fact that the study should be as pragmatic as possible and have a clear focus, which is the investigation of the use of smart functionalities to support the law-making process, investigations on the state-of-play were limited to the most recent relevant material. The same approach was applied in the discussion of LLMs and their eventual utilisation as core elements for building some of LEOS' smart functionalities. Hence, without over-hyping their use, a preliminary screening of the state-of-play in LLM development was performed (*Section 2.5*).

The next section (*Section 3*) outlines the methodological approach employed in the study, comprising three parts. First, it presents the methodology for examining the categorisation of smart functionalities (*Section 3.1*), followed by the methodology to approach their techno-business feasibility (*Section 3.2*). Ultimately, the detailed methodology for evaluating their implementation is given (*Section 3.3*). This analytical exposition of methods is deemed crucial for ensuring the validity of the results and the credibility of the study's recommendations.

After discussing the methodology, the development of smart functionalities categorisation follows (*Section 4*). It begins with an overview of previous categorisation efforts (alpha categorisation), proceeds through intermediate stages (beta categorisation), and then proceeds into the analysis of empirical data collected during the study. The outcome, gamma categorisation, is presented in Section 4.3.

In the subsequent section, five key technologies suitable for implementing a prioritised set of 11 smart functionalities are presented (*Section 5*). Each technology is accompanied by a definition and a technological analysis, followed by an overview of the current state-of-the-art and a selection of open-source solutions. Given the dynamic nature of the AI sector, three representative solutions are highlighted for each technology. These technologies are widely adopted and the underlying algorithms can be seamlessly integrated into LEOS. The necessary integration approach is discussed in *Section 5.6*.

It is followed by *Section 6*, which represents the main outcome of the study and encompasses the examination of the business value and techno-business feasibility of various smart functionalities for LEOS. The business value assessment of specific smart functionalities was derived from the analysis of interviews with EC experts. The section also addresses issues related to deployment, system integration, dependencies, and, more broadly, implementability. The aforementioned examination is followed by a high-level implementation roadmap (*Section 7*). Ultimately, the main conclusions are presented followed by an outlook (*Section 8*).

The report encompassed five appendices: *Appendix I* exhibits the initial roster of smart functionalities, *Appendix II* highlights tools and companies specialising in Legal Data using LLMs, *Appendix III* comprises data from the questionnaire regarding the prioritisation of smart functionalities, *Appendix IV* presents a matrix detailing the attributes of the revised categories, and, lastly, *Appendix V* provides an overview of the technologies linked with the smart functionalities.

2 State-of-play

2.1 The reference study

The reference study titled *Drafting Legislation in the era of AI and Digitisation* is a comprehensive document conducted by the University of Bologna and delivered in 2022, exploring AI's potential in legislative drafting and the digitisation of legal sources.[2] The study, which serves as the starting point for the current one, is divided into three parts:

- **Scope of the study and execution**: This section introduces the analysis, its methodology, and the state-of-the-art in AI and the law. It discusses the benefits of applying AI in the legal domain, the maturity of the market and the conclusions of the explorative research and consultation in the EC;
- **Illustrating the potential of AI and implementation**: This section presents various use cases that demonstrate the potential of AI in legislative drafting. It includes case studies on learning from examining corrigenda, transposition of EU directives, derogations and transitory provisions, and checking for digital readiness. It also discusses the benefits of these use cases, the obstacles encountered and considerations for implementation;
- **Roadmap and recommendations**: The final section provides a roadmap for applying AI in legislative drafting and offers recommendations for its implementation.

This study explores how innovative IT and AI can enhance legal drafting and improve the quality, efficiency and transparency of law-making in the EC. The vision is to enable a paradigm shift through machine computable law, combining advances in IT, the use of standards and progress in understanding law-making theory and practice. A well-integrated IT ecosystem with an 'Augmented LEOS' at its core could transform legislative processes with significant impact.

More specifically the study covers:

- The state-of-the-art in representing legal knowledge, applying AI to law and responsible hybrid AI approaches. Standards like LegalXML and AKN provide machine-readable legal data;
- Use cases demonstrating the potential of AI, like analysing corrigenda to avoid errors, verifying transposition of EU law, detecting derogations and assessing digital readiness;
- Functionalities that could assist drafters, such as context-aware verification, tracking changes, legal assistance in drafting and analysing the legal system;
- Roadblocks like risk aversion, resistance to change, regulatory challenges, data quality and skills gaps. Enabling factors are public sector commitment, partnerships and technological maturity;
- Considerations for implementation architecture, strategy and adopting an open source approach for 'Augmented LEOS';
- A basic implementation roadmap with steps like creating a common dataset, setting up expert task forces, developing prototypes, and building a community of practice;

[2] The reference study: https://bit.ly/3unUAHs

- Recommendations for high-level management support, thoughtful use of AI in law-making, more experimentation and piloting, and the strengthening of cooperation between institutions.

In summary, the reference study provides a solid basis to explore this area further and consider larger-scale pilots and offers a wide view of various technological solutions applied to legal drafting. Moreover, a collaborative IT ecosystem is proposed to harness digital change in legislative drafting, demonstrating the high potential to transform processes and improve quality and efficiency. The reference study resulted in a set of 33 bulk smart functionalities, which were subsequently grouped in categories (alpha categorisation, see *Section 4*).

2.2 The LEOS system and its technology stack

In the course of the current study, the software characteristics of the LEOS tool were investigated. The project team had already assessed the 3.0.0 version of LEOS (Leventis et al., 2021), not only in regards to its internal structure and functionality but also in relation to its integration potential (Fitsilis & Mikros, 2022). On a different, yet equally important note, the training aspects for onboarding the LEOS tool need to be critically assessed, with some aspects already under scrutiny (Fitsilis & Papastylianou, 2023).

For the current purpose, a broad assessment was performed on the system readiness to integrate with and utilise an AI-based set of applications and technologies and/or existing legal or *ParlTech* applications (see *Section 5.6*). This included an investigation of the potential types of integration, such as built in components and API endpoints in the services layer or via external integrated systems interacting with the LEOS tool. Nonetheless, the actual integration of smart functionalities in LEOS may involve the practical investigation of the following key points:

- **API management process for adding integration capabilities**: This is to investigate the procedural aspect of integrating additional external systems that interact with LEOS and add value to its scope;
- **External systems integration**: This is related to the above but centres on the development aspects of system integration;
- **LEOS configuration via API**: Preliminary evaluation shows that a major part of business logic is configured into the LEOS' codebase and requires software engineering work to modify. This is necessary to be exposed via an API;
- **Additional document types**: Additional work may introduce different document types, as well as the creation of document collections, for instance written question and answer documents, which will eventually support the making of an entire document ecosystem (including correlations and hierarchies between documents);
- **GUI embedding of additional or custom features**: New, advanced features will require user input, which might modify and/or add new GUI components.

Exchange with active LEOS end users presupposes an accurate user mapping in order to determine a meaningful set of interviewees/discussants. An external mapping from 2020 is shown by Fitsilis and Makropoulou (2022). This study, however, focuses on EC-internal applications of LEOS, known internally as "EdiT". Throughout its duration, a series of interviews was conducted with EdiT users. In addition, a separate set of consultations might be useful, in order to collect good (and perhaps bad) practices of systems integration and eventual own additions to LEOS' code base.

Talking and addressing the issues around LEOS is one side of the coin. As also highlighted in some of the above issues, at some point, one also needs to assess its integration with other external systems and platforms. For this purpose, the possibility of an integration platform needs to be discussed (Leventis, 2022). To create an integration platform that promotes interoperability, it should be designed as a structured system with a focus on distributed, message-oriented communication and feature exchange. This can be achieved through the utilisation of software agents, which will act as specialised connectors for supported subsystems and serve as interfaces to other systems. These agents will effectively capture the unique characteristics of each system by defining and managing any existing heterogeneity.

Such an integration platform can be structured with the encouragement and facilitation of interoperability at its core in the form of a distributed system that will be realised through software agents, which will develop as specialised connectors for supported subsystems and serve as their interface to the rest of the systems. They will outline each system's unique characteristics by defining any heterogeneity that exists. For addressing the current scope of investigation, software agents can be enhanced with AI features and more autonomy.

Additionally, it is recommended to make the platform readily accessible through a software-as-a-service model, ensuring a low barrier of entry. This approach would promote its adoption within any governance workspace, while also providing incentives for commercial applications. In this context, it is crucial to note that LEOS is accessible on code.europa.eu,[3] which serves as the code development platform for open-source projects shared by the European Union institutions. The scrutinised LEOS release did not consider strong authentication or access controls schemes. It is considered imperative for an enhanced architecture to support additional security or authentication protocols.

2.3 AI in the public sector

AI encompasses a wide range of methods, patterns, and approaches designed to replicate and approximate human-like behaviour when solving complex problems (Winston, 1992). With rapid advancements in AI, associated tools and services are becoming increasingly mature, gradually making their way into the public sector (Van Noord & Misuraca, 2022). AI-based applications include, for instance, sophisticated tools for citizens' engagement in political debates and the law-making process. These technologies hold the potential to revolutionise governance institutions, transforming both the production and delivery of public services.

[3] code.europa.eu platform: https://code.europa.eu/

There are obvious potential use cases for LEOS in these operational environments. *Table 2.1* displays a selection of such use cases and relevant studies, both from the *executive* and the *legislative* power. These refer to a wide range of overlapping fields of science and technology, such as machine-readable law, LegalXML, hybrid AI and others.

Scholars	Use Case
Applying regulatory patterns	
Levagin et al., 2022	Simplified procedure for interactively estimating compliance costs when introducing new legal provisions using machine learning. Ongoing study by the Federal Statistical Office, Germany (in German)
Micheler & Whaley, 2020	Regulatory Technology: Replacing Law with Computer Code; Distributed Ledger & AI technologies are discussed for regulatory applications; the regulator will need to retain a substantial amount of oversight over its design in order to retain legitimacy & accountability.
Senninger & Blom-Hansen, 2020	Analysis of the EU Commission's Regulatory Scrutiny Board through quantitative text analysis. The authors studied 673 Board opinions and compared <100 draft and final policy proposals using machine learning techniques and quantitative text analysis.
Using Legal XML	
Palmirani, 2022	Hybrid AI to Support the Implementation of the European Directive; research within the framework of the reference study.
Ma & Wilson, 2021	The legislative recipe: syntax for machine-readable legislation; logic syntax and symbolic language; the limits and challenges of machine-readable vis-à-vis human readable legislation.
Transforming Law-Making in the EU	
Minghini et al., 2022	(Special issue) A European Approach to the Establishment of Data Spaces; development and implementation of European strategy for data; build a single data market by establishing a common European data space.
Utilising AI in parliaments	
Fitsilis & de Almeida, 2024	Collection discussion, and classification of 39 use cases of AI tools and services in the parliamentary workspace
von Lucke et al., 2023	Identification of 210 potential uses for AI in the parliamentary workspace classified by sectors
Moschopoulos, 2023[4]	AI applications in EP administration - current status; listing and describing AI apps in production: chatbots, speech-to-text, automatic document indexer and low-code analytic tools.
Fitsilis, 2021	AI in parliaments – preliminary analysis of the Eduskunta experiment; use and interaction with a GPT-2 based chatbot by the Committee for the Future of the Finnish Parliament.
Agnoloni et al., 2022a/b	Making Italian Parliamentary Records Machine-Actionable; clustering Similar Amendments at the Italian Senate.

[4] AI in the European Parliament administration: https://bit.ly/44HCMn4

ittig[5]	Identification of references to European and Italian laws, and other types of acts, in a given text fragment; automatic mark-up of each reference as a hyperlink.
European Parliament, n.d.[6]	European Parliament archives dashboard; text summariser for the European Parliament archive documents, reducing the original size and facilitating EU citizens research.
AI in government	
Pathak et al., 2021	Topic-level sentiment analysis of social media data using deep learning. It can be used at several stages of the policy cycle to extract user opinions and stances in almost real time.
Alexopoulos et al., 2019	Machine Learning (ML) influencing e-Government. Comprehensive analysis of the use of ML by governments. Findings contain potential benefits & barriers.
Criado & Gil-Garcia, 2019	Creating public value through smart technologies and strategies: From digital services to artificial intelligence and beyond.

Table 2.1. Selected use cases of AI in the public sector.

The focus was primarily put on use cases from Member States and EU institutions and agencies, with an exception for a UK-based solution. Language is an issue when assessing the state-of-play as national studies might not be captured. The project expert mainly studied materials in English, with the exception of a German study on the use of machine learning for regulatory impact assessment (RIA). As LEOS is developing to become an application/platform of wider scope and geographical relevance, it is maybe necessary to expand the review in use cases from around the globe. This, however, evades the scope of the current investigation and could be conducted at a later stage, e.g. in the course of a business development study.

There are several contributions reviewing the considerable volume of literature or proposing various classification frameworks for *LegalTech* (Mania, 2022; Harpe & Zhang, 2021). However, only a handful of these are concerned with actual solutions that can be associated with smart functionalities and/or specific use cases. What is evident on several occasions is the fact that the Akoma Ntoso (AKN) standard is used to perform a series of tasks, such as comparison of articles, extraction of definitions and identification of normative citations. In this regard, it needs to be mentioned that LEOS uses the Akoma Ntoso for the European Union (AKN4EU) common formal for legislative documents.[7] AKN4EU is based on the IMFC Common Vocabulary[8] and constitutes a machine-readable structured format for legal document exchange within the EU decision-making process.

Some of the most useful reads have to do with the application of AI in the parliamentary workspace. While this is not surprising *per se*, it can be explained by the fact that they were mainly drafted by practitioners rather than scholars, therefore centring on digital solutions

[5] ittig: The Institute of Legal Information Theory and Techniques; see https://linkoln.gitlab.io/ and https://www.normattiva.it/

[6] AI in the European Parliament Archives: https://bit.ly/43nwtE5

[7] AKN4EU: https://op.europa.eu/en/web/eu-vocabularies/akn4eu

[8] IMFC common vocabulary: https://op.europa.eu/en/web/eu-vocabularies/cov

instead of theoretical constructs and general frameworks. Such parliamentary applications include a lot of cognitive AI (thus not immediately associated with LEOS), but also a significant number of Natural Language Processing (NLP) based tools and services, such as speech to text, word prediction, text classification, semantic similarity, chatbots and others. These all, again, are applications that can be potentially associated with LEOS.

Moving on to the core subject of the present study, while the dynamics of new technology become evident in the surrounding tasks, such as in the making of regulatory impact assessments (RIAs) or in AI-enhanced policy making,[9] it seems that only limited effort is placed in the actual drafting of legislation. In the authors' view, this appears to be the case not only within the EU but also on a global scale. This can be attributed to the fact that in the recent past significant resources were invested in more basic digitisation and/or digitalisation tasks, such as Optical Character Recognition (OCR), markup languages and machine-readable formats and standards, system and interoperability specifications, strategic planning, ethical and operational guidelines, and others.

The above assessment, however, does not take into account the evolution of versatile open source legislative tools, such as LEOS, and the rapid development around LLM's. The former offers a solid base upon which any given legal order may integrate numerous functionalities around the legislative process, while the latter might very well form the technology core for the implementation of a large number of the smart functionalities that were already identified by the reference study.

2.4 Legal linguistic resources

Given its specialised nature, legal language demands dedicated resources for study and application. These resources encompass corpora, lexical databases, grammatical and stylistic guidelines, as well as references for acronyms, organisations, and abbreviations. The legal linguistic resources originating from the EU are formatted in italics.

Corpora

Corpora are collections of texts that can be processed using specialist software. They are essential for examining words and phrases in their contexts and comparing them in different periods. Some notable legal corpora include:

- The United Nations Parallel Corpus[10] and the *Digital Corpus of the European Parliament* (DCEP),[11] which are publicly available legal corpora created under the auspices of international organisations like the UN or WTO;

[9] Substantial work has been conducted in the past decade in the collection, handling and analysis of citizen data (particularly user sentiments) from digital platforms, especially social media - a vast field of application of AI, from which policy making can benefit.
[10] UP Parallel Corpus: https://conferences.unite.un.org/uncorpus
[11] DCEP Digital Corpus of the European Parliament: https://bit.ly/3OtAIJL

- The Corpus of Contemporary American English (COCA)[12] and the Corpus of Historical American English (COHA)[13] contain a variety of texts, including legal ones;
- The Law and Corpus Linguistics Technology Platform, which offers a user-friendly interface for searching corpora by terms;
- The SOULL (Sources of Language and Law) platform,[14] which provides information about existing data collections and corpora of legal language in various sizes, languages, and text types.

Lexical Resources

Lexical resources are essential for understanding the specific terminology used in legal language. Some of these resources include:

- Black's Law Dictionary,[15] a comprehensive open online law dictionary;
- Garner's Dictionary of Legal Usage (Garner, 2001).

Grammatical and Stylistic Guides

Grammatical and stylistic guides provide rules and recommendations for writing in legal language. Some of these guides include:

- The Legal English Grammar Guide (Davies, 2020), which guides the user through important tenses and aspects of the language;
- The Adobe Legal Department Style Guide,[16] which provides guidelines for clear legal writing and enhanced collaboration on document creation;
- The Grammar and Writing Handbook for Lawyers (Espenschied, 2011), which shows precisely which rules need to be followed, how to choose the correct words, and the most effective way to structure every sentence.

Resources for acronyms, organisations and abbreviations

Multiple such linguistic resources for acronyms, organisations and abbreviations already exist. Some of these are listed below:

- *Eurostat's List of Abbreviations and Acronyms*:[17] This list includes abbreviations and acronyms used in the context of the European System of Accounts and other EU-related terms, such as EC (European Commission), ECB (European Central Bank), and EMU (economic and monetary union);

[12] COCA: https://www.english-corpora.org/coca/
[13] COHA: https://www.english-corpora.org/coha/
[14] SOULL: https://legal-linguistics.net/data-collections/
[15] Black's law dictionary: https://thelawdictionary.org/
[16] Adobe Legal Department Style Guide: https://www.adobe.com/content/dam/cc/en/legal/documents/ADOBE-LEGAL-STYLE-GUIDE.pdf
[17] Eurostat: https://ec.europa.eu/eurostat/esa2010/chapter/view/27/

- *YourDictionary's List of EU Abbreviations*:[18] This resource provides abbreviations for languages within the EU, as well as abbreviations for EU member countries and various EU organisations and programs;
- *fi-compass Acronyms*:[19] This list includes acronyms related to European Structural and Investment Funds, such as ESF (European Social Fund) and ESIF (European Structural and Investment Funds);
- *Publications Office of the EU* / Main Acronyms and Initialisms:[20] This annex lists acronyms and initialisms used in EU documents, including those related to various EU agencies and policies;
- *Clingendael's List of Abbreviations and Acronyms*:[21] This list includes abbreviations and acronyms used in the context of the EU's role in promoting democracy or stability, such as EP (European Parliament) and EULEX (European Union Rule of Law Mission in Kosovo).

In the following section, the intersection of Large Language Models (LLMs) and the legal domain is presented.

2.5 LLMs and legal data

In the course of the study, a comprehensive list of LLMs was put together that have been fine-tuned on legal corpora, showcasing the remarkable strides made in adapting these powerful models to the legal field's unique linguistic and conceptual complexities. Furthermore, a variety of companies and tools were explored that leverage these fine-tuned LLMs to process and analyse legal data, demonstrating the practical applications and transformative potential of these technologies in real-world settings.

To provide a more granular perspective, these applications were broken down by domain, highlighting the diverse ways in which LLMs are being utilised across different areas of the legal landscape. This short survey aims to provide a snapshot of the current state of LLMs in the legal field, offering insights into the ongoing advancements and future directions of this exciting interdisciplinary frontier. While the field undergoes continuous updates, this compilation incorporates the most significant contributions available up to mid-July 2023.

Appendix II breaks topics down according to the area of application, gives the name of the company and/or the product offered. Each entry contains a short description with particular reference to the foundational LLM used and information related to the functionalities implemented. Moreover, the last column gives a link to the webpage of the entry. The whole list is organised in various application areas to facilitate the navigation of the reader to the technological solutions provided.

[18] YourDictionary: https://www.yourdictionary.com/articles/eu-abbreviations
[19] fi-compass: https://www.fi-compass.eu/info/acronyms
[20] Publications Office of the EU: https://publications.europa.eu/code/en/en-5000400.htm
[21] Clingendael: https://bit.ly/3SvjeO9

The following table contains some of the most well-known LLMs that have been finetuned to the legal language (see *Table 2.2*). More specifically, the first column contains the name of the finetuned version of the LLM. The second column gives information about the language that the LLM can handle and also the foundational LLM used for finetuning. The last column contains a link to the paper that has introduced the LLM. In some cases, the paper contains also the link to the GitHub repository of the finetuned version of the LLM.

Name	Language(s) / Base LLM	Link to Paper
LawFormer	Chinese / LongFormer	https://www.sciencedirect.com/science/article/pii/S2666651021000176#bib38
LawyerLLaMA	Chinese / LLaMA	https://arxiv.org/pdf/2305.15062.pdf
BERT finetuned in LeNER-Br corpus	Portuguese /BERT	https://link.springer.com/chapter/10.1007/978-3-030-61377-8_46
BudgetLongformer	English / LongFormer	https://arxiv.org/pdf/2211.17135.pdf
CrimineIBART	French / BART	https://dl.acm.org/doi/10.1145/3462757.3466147
ITALIAN-LEGAL-BERT	Italian / BERT	https://ceur-ws.org/Vol-3256/km4law3.pdf
LEGAL - BERT	English / BERT	https://aclanthology.org/2020.findings-emnlp.261/
CaseLaw BERT	English / BERT	https://arxiv.org/abs/2104.08671
PoL-BERT	English / RoBERTa	https://arxiv.org/pdf/2207.00220.pdf
LexLM	English / RoBERTa	https://arxiv.org/pdf/2305.07507.pdf
InstructLAW	English / GPT3.5	https://blog.servient.com/blog/instructlaw

Table 2.2. Recently developed LLMs finetuned to legal corpora.

The above lists are continuously expanded taking into consideration user experience and reinforcement learning from human feedback (RHLF). Also, evolution in the next time is expected to be driven by the following trends:

- Qualitative-driven training corpora compilation;
- Reducing parameter space and exploiting it in a more efficient manner;
- Expanding the context of the word embeddings included;
- Multi-modal training for enhanced knowledge representation.

For LLMs in the legal-technical domain, the above might have considerable gains, such as in the interpolation of existing knowledge base for better approximating the target policy issues. More specific, ready-to-use legal provisions are, hence, possible to be generated.

2.6 Synopsis of current situation

A lot of work has been conducted to introduce state-of-the-art technology to the usually conservative legal domain. In the meantime, countless approaches and even more use cases may be found in the literature. The LEOS tool is just one of them. Hence, advancing LEOS by using so-called ‘smart functionalities’ will not be a straightforward task.

For capturing the state-of-play in science and technology towards an “augmented LEOS”, a broad literature and AI-based tools assessment has been first conducted. Preliminary results show that recent advances around LLMs present numerous *realistic* implementation opportunities. Considering the overall steep development curve around the investigation of such operational and procedural patterns, it is to be expected that the situation might shift in the near future. Hence, any absolute statements and outcomes must be avoided.

Similarly, any software products implementing functionalities based on disruptive technologies need to be developed using an agile rather than the typical waterfall approach, while keeping an eye on changing technology stacks. This is why the integration patterns of such add-ons will be of importance. Integrating smart functionalities directly to LEOS, e.g. using a dedicated API, might be an option, however it might be more useful to broadly address the issue of integration of the entire tool on various platforms.[22] Such platforms could as also function as a smart functionalities repository, thus creating in essence an entire ecosystem for such ‘smart’ solutions.

At the same time, the present analysis shows that it is of particular significance to operate within a well defined operational and ethical framework. For the case of utilising smart add-ons within the parliamentary workspace, the relevant AI guidelines need to be seriously taken into consideration (Fitsilis et al., 2023). Following a relevant assessment of the state-of-play in any given institutional setup (Koryzis et al., 2021), a subset of these guidelines could be introduced even before the actual implementation of smart functionalities. The guidelines also include issues of capacity building of the experts that will be dealing or will be influenced by the operation of smart tools and services. Hence, assuming that humans will be the major enabling factor, human-centric software development,[23] legal/ethical legitimation and inclusive training of the stakeholders to the law-making process will become essential steps towards the introduction and evolution of such tools.

[22] See, for instance, code.europa.eu and Joinup.

[23] See the human-in-the-loop, human-on-the-loop and human-in-command approaches for designing AI-based systems.

3 Methodology

3.1 Method of categorisation of smart functionalities

Categorisation of smart functionalities in LEOS is crucial for a number of reasons. Firstly, it enhances the end user's understanding by grouping similar features. Secondly, it aids in an effective software planning process, ensuring a clear roadmap for feature development and maintenance. On the same note, it also facilitates prioritisation, allowing focusing on essential functionalities and allocating resources appropriately. Ultimately, it contributes to better user experience, streamlined development processes, and the overall success of the software. The current work took into account the investigation and analysis already conducted in the course of the reference study. Apart from the authors' own expertise in the design, installation, and operation of LEOS, this study gathered and evaluated substantial empirical data through a combination of questionnaires and structured interviews.

Specifically, a set of eleven interviews and a technical workshop on LEOS were conducted between 25 July 2023 and 8 September 2023. *Table 3.1* below lists the European Commissions' Directorates-General (DGs) from which experts were interviewed. Substantial care was taken to interview experts from as many DGs of the European Commission as possible, in order to obtain rich information on the user experience with LEOS in various use cases and implementations. Moreover, a technical workshop on LEOS was conducted on 27 July 2023 with the goal of tapping into insider knowledge and development goals for the state-of-play and the evolution of the software. Additional one-to-one meetings or chats with developers and architects via an established MS Teams channel were used to determine details of the design and implementation of the LEOS system.

The interviews followed a structured, predefined format, and their duration was limited to 60 minutes. The agenda included a broad presentation of the project and the contractor's experts, a description of the experience of the interviewed person(s) with the LEOS/EdiT system, and a reflection on the potential and challenges arising from the use of smart functionalities. The use of Large Language Models (LLMs) was often brought into the discussion. Following each interview, the preliminary list of smart functionalities that was determined in 2022 by the reference study was shared with the interviewees.[24]

Through this form of the questionnaire, the interviewees were asked to indicate five smart functionalities without prioritisation that they would like to see implemented first, while providing a brief rationale for each selection. Desired smart functionalities off the list could also be included in their selection, with a brief explanation. The aforementioned approach constitutes, in principle, an expert survey.

[24] This list includes an additional smart functionality compared with the 33 included in the reference study. This is functionality #26 - *Large Language Model (LLM) based legal text generation*.

#	DG	Date
1	JRC-SEVILLA	25/7/2023
2	REGIO	26/7/2023
3	SG	9/8/2023
4	DGT	16/8/2023
5	FISMA	17/8/2023
6	SJ	30/8/2023
7	JRC-ISPRA	31/8/2023
8	DGT	1/9/2023
9	SJ	6/9/2023
10	OP	8/9/2023
11	DIGIT	8/9/2023

Table 3.1. List of interviews.

In expert surveys, which involve special and limited populations, the sample size is intentionally small and there is no need for a representative sampling framework. In this study, we employed purposeful sampling for data collection, emphasising predominantly qualitative interpretation.

The main points from the interviews, including the experts' suggestions for smart functionalities for an augmented LEOS/EdiT instance are discussed in *Section 4.2*. The project experts utilised these exchanges as well as the questionnaire to determine the categories of smart functionalities. The collection of rich empirical data and their subsequent analysis according to a well-defined feature scheme for each cluster of smart functionalities, ultimately allows for a more flexible and goal-oriented approach. Also, gradually adding more smart functionalities to the original set may eventually lead to:

- enhanced understanding of the trends driving legal drafting at the EC level;
- conceptualization of potentially more diverse categories of smart functionalities;
- headstart towards developing a more coherent and detailed final report.

The presentation and discussion of the categorisation of smart functionalities took place in Brussels on 25 September 2023. Additional input and comments were also collected during the SEMIC 2023 conference in Madrid on 17-18 October 2023.[25]

[25] SEMIC 2023 conference: https://semic2023.eu/

3.2 Method of approaching the techno-business feasibility of smart functionalities

The evaluation of the techno-business feasibility of smart functionalities was carried out across three tiers, in a quasi-parallel fashion. The initial tier involves assessing and documenting the **business value**. To achieve this, a group of European Commission professionals was interviewed between August and September 2023. A questionnaire featuring pre-identified smart functionalities was also circulated, and the responses were meticulously evaluated. The second tier focuses on the **technical feasibility** of the identified functionalities. This aspect necessitates a preliminary framework due to its potential breadth. Here, the investigation aimed to assess both the technology and the approach that are necessary to integrate a selected subset of smart functionalities into LEOS, while exploring possible means and prerequisites for implementation. This exploration encompasses considerations such as data and knowledge requirements, as well as the sources and quality of information.

The means of implementation inherently encompass a technological dimension that requires precise specification. The suggested approach involves evaluating potential algorithms and/or technologies, whether AI-based or not, for the implementation of each smart functionality (see also *Section 3.3*). This assessment extended to the evaluation of relevant applications and tools that could be seamlessly integrated into the system. Given the nature of LEOS, it was imperative to investigate the open-source dimension of such tools. Following the determination of these aspects, the overall suitability of the LEOS architecture was subject to critical discussion. This discourse extended to housing, hosting, and infrastructure considerations.

In the third tier of analysis, the overall **business feasibility** of smart functionalities was explored and assessed. By combining insights from the initial two tiers, various associated parameters were discussed. To facilitate a clear and straightforward organisation of the discussion on these steps, a set of five defining attributes has been established: UX, business value, technology stack, data sets and performance. Each of these attributes are discussed for each of the specific smart functionalities under scrutiny.

3.3 Method of assessing implementation of smart functionalities

The study primarily concentrates on smart functionalities frequently selected by EC experts. A significant path for their implementation involves clustering technologies based on these selections. This approach was chosen for two primary reasons: to use resources more efficiently and to speed up innovation and development.

When it comes to technology, clustering of various technological solutions and efficient utilisation of resources is paramount. By grouping similar technologies, the European Commission can streamline its development efforts, reduce redundancy, and avoid duplicative work. This approach enables shared learning, common infrastructure, and collaborative problem-solving, ultimately maximising the impact of resources invested in the development of AI-based tools and services, such as the envisaged LEOS smart functionalities. The core technologies for each SF in the LEOS editing platform were identified and selected based on their relevance and efficacy in addressing specific challenges within legal, governmental, and parliamentary document management. This process involved a thorough analysis of each functionality, considering the unique demands of legal text processing.

For the most upvoted SFs five core technologies were identified that will be elaborated further in *Section 5*:

- *Semantic Similarity* was chosen for tasks requiring nuanced understanding of language and context, essential for correlating different sections of legal documents and detecting hidden semantic correlations.
- *Named Entity Recognition* was applied to functionalities involving the identification of specific legal entities, citations, and references, owing to its precision in extracting structured information from unstructured text.
- *Information Extraction* was designated for functionalities where extracting specific data, such as obligations, rights, and legal statuses, from complex legal texts is crucial.
- *Natural Language Generation* was assigned to tasks requiring the generation of new legal text, like drafting amendments, due to its ability to produce coherent, contextually appropriate content.
- *Advanced Language Editing and Correction* was selected for functionalities that necessitate sophisticated linguistic and stylistic refinement of legal texts.

The technology clustering that was applied separates the prioritised smart functionalities into five distinct groups, i.e., I to V in Roman numerals, as per *Table 3.2*. Notably, the smart functionalities discussed are not uniformly spread among the five technology clusters. Specifically, clusters I and III each encompass four functionalities, while the remaining three clusters house only one smart functionality each. These are already established and tested technologies, whose underlying algorithms can be embedded in LEOS, for instance through user experience add-ons and extensions and their respective APIs, which will provide the backend functionality through a common integration platform.

Cluster	Technology	Smart Functionalities
I	Advanced Language Editing and Correction	#9-#10-#12-#13
II	Named Entity Recognition	#3
III	Semantic Similarity	#11-#14-#15-#20
IV	Natural Language Generation	#26
V	Information Extraction	#19

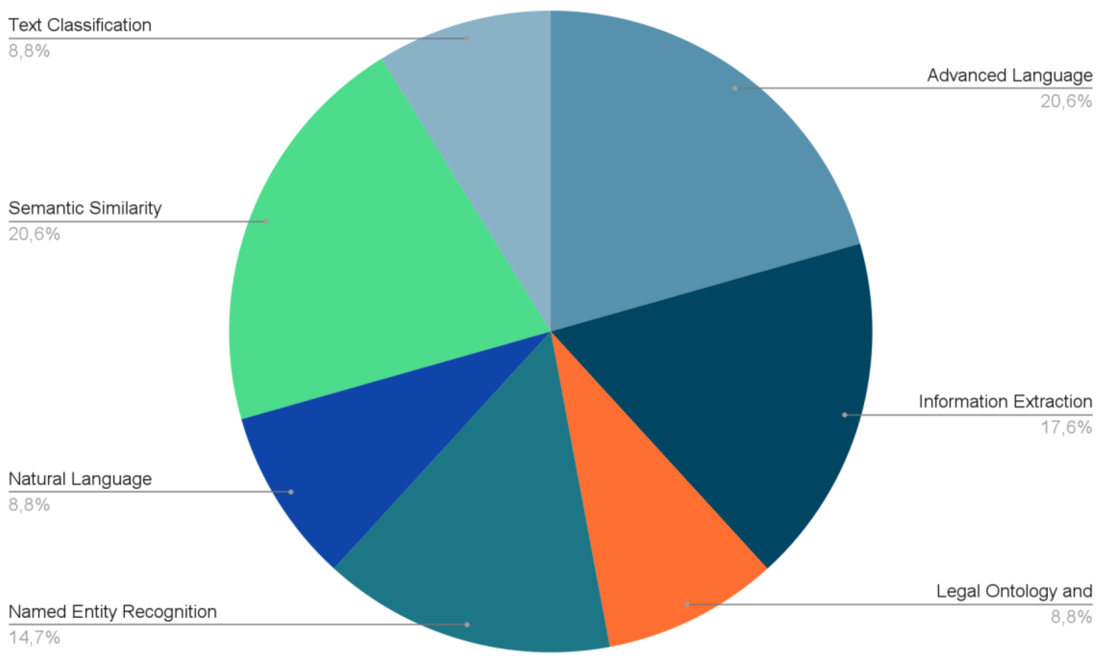
Table 3.2. Clustering of smart functionalities in broader technology labels.

This systematic approach ensures that each technology is optimally aligned with the specific requirements of the corresponding functionality, thereby enhancing the overall efficiency and effectiveness of the LEOS platform in managing legal documents. Moreover, the clustering of technologies fosters a synergistic environment for similar ideas and innovations. Grouping related technologies enhances the likelihood of knowledge transfer, shared insights, and collaborative advancements. This, in turn, potentially expedites the overall pace of innovation within the clustered technologies, resulting in more agile development cycles and the creation of more resilient AI solutions.

Appendix V links all 34 smart functionalities with their respective core technology. Accordingly, 27 out of 34 (79.4%) smart functionalities can be implemented using the above technologies. In addition, two more core technologies were identified for the rest of the SFs that have collected less than four votes. These were:

- *Legal Ontology and Terminology Management* was applied to tasks involving the management of complex legal terminologies and definitions, ensuring consistency and accuracy;
- *Text Classification* was chosen for functionalities that require categorization and analysis of legal documents based on their content and structure.

Each of these additional technologies serves as the central element for three smart functionalities. *Figure 3.1* depicts the distribution of the full set of seven technologies across the



entire landscape of smart functionalities.

Figure 3.1. Distribution of technologies for the entire SF landscape.

Within the scope of this research, however, detailed examination of the final two technologies will not be conducted. Instead, the current study deliberated on and assessed each attribute for all 11 prioritised smart functionalities. The results are presented in *Section 6*. *Table 3.3* describes the attributes that will be used in the analysis. The following abbreviations will be used for the attributes: user experience (UX), business value (BV), technology stack (TS), data sets (DS), and performance (PER). The category (CAT, gamma categorisation) of each smart functionality (SF) will also be used in the following. During the discussion on the implementation of SFs, attributes with shared characteristics, such as TS, DS, and PER, will be handled horizontally.

Attributes	Description
UX	The discourse on this attribute seeks to proactively pinpoint potential user satisfaction issues, set benchmarks for success, and guide development decisions to optimise the user experience (UX). The ex-ante evaluation for UX specification is grounded in interviews and questionnaires, which have already revealed SF-specific UX components. The analysis indicates a compelling need for a high degree of customisation to meet diverse user needs, given that identical smart functionalities may be required for different purposes within a single or multiple DGs. For example, customisable UX templates could accommodate guidelines or accustomed experience in certain DGs, additional DG-level data sources or processes that should be imported or otherwise included. Several EC experts have identified specific UX features, akin to those found in standard commercial applications (e.g., MS Word). Smart templates, in the sense of pre-designed documents that incorporate intelligent features, such as automation, dynamic content, or interactive elements, are thought to streamline and enhance UX.
BV	Essentially, the BV of any given SF may constitute the primary rationale for its eventual implementation. The overall value is determined by considering and weighting various factors, such as the quality, user effort, automation and efficiency gains, explainability of results, complexity, and others. These factors will be scrutinised in more detail for individual smart functionalities. The insights gained from interviews and questionnaires provided comprehensive information on the business value of the highlighted functionalities. Ideally, a full-scale BV assessment should proactively explore the alignment of the software service with strategic business goals, potential return on investment (ROI), and its overall impact on organisational objectives. However, such in depth analysis lies outside the scope of the present report.
TS	The discussion of TS is designed to proactively evaluate the appropriateness and effectiveness of the technology components that integrate smart functionalities into LEOS. The selection or adaptation of technologies is contingent upon the need to introduce new features without causing regression errors, which could potentially disrupt or impede existing functionalities. In addition, functional aspects such as quality and accuracy of results should also be reviewed and validated in a tantrum with non-functional implementations, for example performance, operational stability, scalability, security, maintainability, and ease of integration. Depending on the specific features, a set of components spanning architectural layers may need to be employed, thereby influencing the UX, API, data layers, as well as the existing algorithms and infrastructure. The centre of weight in implementation is placed on open-source solutions. Any open-source component should be properly documented and thus maintainable.

DS	In developing a software platform for legislation-related purposes, such as the one envisaged around the augmented LEOS, it is imperative to carefully consider diverse datasets for optimal structuring. Effective structuring involves accessing and searching through multiple repositories housing legal texts and pertinent documents on legislation structure, often based on various criteria. Additionally, jurisdictional data is essential for supporting laws specific to particular geographical regions. The platform's capability to access historical data and facilitate cross-referencing between different cases and court rulings is crucial. Furthermore, another critical consideration is securing access to repositories from third parties and organisations, potentially containing valuable data. In developing smart functionalities, it is advisable to initially prioritise attention on EU law and jurisprudence, given the accessibility and quality of relevant data sets. Expanding beyond this scope, for instance to encompass member state or international law, introduces considerable challenges, predominantly stemming from issues related to the quality and homogeneity of decentralised, non-standardised data sets.
PER	When implementing software tools and features, it is imperative to address non-functional requirements, particularly in scenarios involving large distributed datasets requiring access and processing. The overall system performance is susceptible to compromise due to the substantial workload imposed by such implementations. Consequently, critical considerations must be taken into account to mitigate and manage this risk, ensuring the availability of features without detriment to the system as a whole. Performance analysis extends beyond technical aspects and can encompass user engagement, quality of suggestions, legal consistency and coherence, session duration, and feature usage patterns to evaluate overall satisfaction. To ensure reliable tracking and evaluation by system developers, performance indicators must be specific, measurable, and relevant to the specific SFs to which they apply.

Table 2. Attributes of smart functionalities.

Following an intermediate step that deals with the categorisation of the smart functionalities, the high-level design of the top smart functionalities will be provided, as well as a set of tangible recommendations for their implementation.

4 Categorisation

4.1 Previous work on categorisation

The reference study resulted in 33 bulk smart functionalities that were originally summarised and categorised in nine clusters/categories that tacked both the hard core of legal drafting (see, for instance, *Support 'automatic' legal drafting*), as well as preparatory or supporting works, such as various content/context verification and linguistic tasks. The original list and categories of smart functionalities are shown in *Appendix I*. These categories are listed below:

1. **Verification I**: Correct usage of citations, existing references, and other abbreviations;
2. **Verification II**: Context-aware correct usage of references, existing legal definitions, and specific legal lexicon;
3. **Granular change tracking**: Comparison of documents;
4. **Linguistic support**;
5. **Legal Assistance I**: within the act;
6. **Legal Assistance II**: within the legal corpus;
7. **Support 'automatic' legal drafting**;
8. **Policy dimension**;
9. **Discovery of practices**.

The clustering described above is termed "alpha categorization." It comprises two distinct sub-clusters each consisting of two similar categories, one for various *verification* types and another for *legal assistance*. Additionally, it encompasses a range of categories dedicated to aspects of legal drafting, including detailed change tracking, linguistic assistance, and the identification of both exemplary and potentially good (or problematic) practices. Lastly, within the alpha categorization, there are specific categories focused on policy and automated legal drafting, which are closely linked to the fundamental aspects of law-making.

The above clustering offers a logical structure and includes categories of smart functionalities that potentially save a lot of time by allowing drafters to concentrate on substance matters. *Granular change tracking* facilitates collaborative drafting, the very essence of LEOS' functionality. *Verification* and *linguistic support*, ease the life of operators and national authorities, for instance, by avoiding litigation resulting from textual and linguistic divergences. The *discovery of practices* could help with the drafting of provisions, which are usually similar, allowing the drafter/reviser to adapt previous (good examples) to the circumstances of the case.

The questionnaire that was circulated included this exact alpha categorisation, with an important addition. As already indicated above, a new item (#26: LLM-based legal text generation) under the category *support 'automatic' legal drafting* has been placed in the questionnaire that has been circulated to the EC's experts. Prior to the initiation of the current study, the above categories were subsequently refined as follows ("*beta categorisation*"):

1. **Context-aware legal verification**: accurate usage of citations (validity and relevance), references exist and are up to date, consistency of definitions, correct use of specific lexicon, acronyms and other abbreviations;
2. **Fine granular change tracking**: comparison of documents, modifications, change tracking, revision features, and smart assisted merge of different versions;
3. **Linguistics support**: use correct linguistic formulations within the structure of the document, correct formulation in accordance with English Style Guides, detect divergences between different linguistic translations, suggest linguistic formulations in provisions;
4. **Legal assistance – within the act and between acts**: for instance, detect and avoid structures that could create unintended ambiguities in legal interpretation, correlation between the recitals and the enacting terms, linkages between preceding acts and the one being drafted, identification of incompatibilities in temporal parameters, detection of explicit or implied obligations, highlight rights, permissions or penalties;
5. **'Automatic' legal drafting**: for instance, for drafting transitional measures, the use of templates on regulatory reporting requirements or constructing the consolidated text applying amendments;
6. **Input on policy dimensions**: for instance, estimate the impact of a legislative act, measure the digital-readiness or contribution to gender-equality strategy;
7. **Discovery of legal drafting practices**: detect patterns, good practices, and common errors;
8. **Advanced visualisation:** smart visualisation of legislative content to help the comprehension, positioning, and standing of a legal act, for instance, by clustering acts in meaningful categories and show dependencies.

This refinement was necessary as the EC's understanding of the influence of artificial intelligence in the field of law-making developed. Three main differences between the two categories, alpha, and beta, can be determined:

- The two verification categories were merged;
- The two legal assistance categories were merged;
- A new category (*advanced visualisation*) was added.

As a result, the beta clustering includes eight categories. This categorisation is considered the baseline categorisation for the current report. In the following sections, these categories are to be further refined, and the functionalities therein re-classified, based on the insights earned from the stakeholder exchanges and the collected questionnaires.

4.2 Analysis of empirical data

4.2.1 Questionnaire results

The current subsection presents and discusses the results obtained from the distributed questionnaires. The response rate reached 91% (ten out of eleven questionnaires received). Most of the respondents provided lists of desired functionalities, with some of them suggesting more than five items. However, some respondents indicated entire categories rather than individual options. In this case, the project experts decided to incorporate into the analysis all smart functionalities that each suggested category includes. *Appendix III* shows the frequency of responses per questionnaire item (smart functionality).

The frequency of selected items is of particular interest:

- All items (34/34; 100%) were selected by at least one expert;
- 27 out of 34 items (79.4%) were selected by at least two experts;
- 16 out of 34 items (47.1%) were selected by at least three experts;
- 11 out of 34 items (32,4%) were selected by at least four experts;
- 2 out of 34 items (5,9%) were selected by at least five experts;
- 2 out of 34 items (5,9%) were selected by at least six experts.

The distribution of the findings can also be plotted as a graph (see *Figure 4.1*).

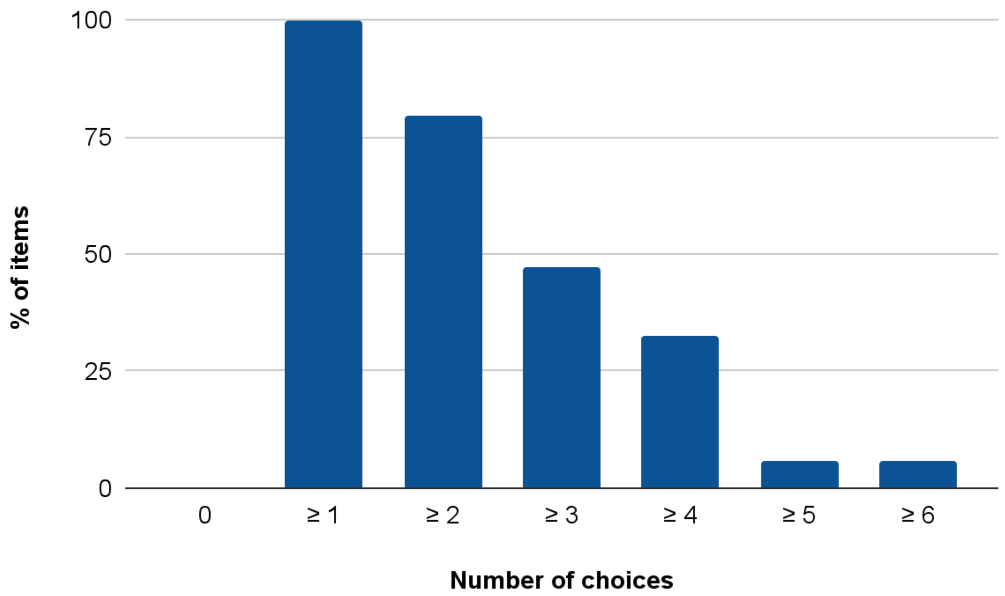


Figure 4.1. Distribution of the EC experts' choices.

Seeing the above distribution, it becomes clear that all items from the original list of smart functionalities are relevant. This rather speaks for the high relevance of the proposed items from the reference study. Then, almost half (47,1%) of the items were opted by at least three experts (out of 10: 30%) that responded to the questionnaire. This figure is perceived to be high and shows that still a notable number of smart functionalities is considered useful for a significant number of EC experts, further supporting the previous assumption regarding the high relevance of the vast majority of items.

SF	Title	No. of picks
#14	Correlation between recitals and the enacting terms	6
#20	Automatically identify existing legislation relevant for the act under development	
#3	Acronyms, organisations and other abbreviations	4
#9	Use correct linguistic formulations within the structure of the document	
#10	Correct formulation in accordance with the English Style Guide	
#11	Detect divergences between different linguistic translations	
#12	Suggest linguistic formulations in provisions	
#13	Detect and avoid structures that could create issues in legal interpretation	
#15	(Correlation) between previous acts and the new one	
#19	Detect obligations, rights, permissions, penalties	
#26	Large Language Model (LLM) based legal text generation	

Table 4.1. Most ‘picked’ smart functionalities.

Table 4.1 shows the 11 smart functionalities that were selected by at least four experts. Understanding the preferences and acceptance of these items among LEOS stakeholders or end-users can be crucial when determining suitable candidates for an eventual Proof-of-Concept phase. Evidently, a couple of items display a remarkable popularity as they were chosen by at least half of the responders. These are #14 (four picks): *Correlation between recitals and the enacting terms* and #20 (five picks): *Automatically identify existing legislation relevant for the act under development*, both of which belong to the *Legal assistance* category (beta). What appears to be interesting is the fact that the lately introduced smart functionality on LLM-assisted law making (#26) does not belong to the top picks, something that deserves more investigation. When studying the distribution of expert choices according to beta categorisation, a series of interesting features stand out (see *Figure 4.2*).

There are three categories that account for almost three fourths of the favourable items (71.5%): *legal assistance*; *legal verification*; and *linguistic support*. In particular, the smart functionalities performing legal assistance tasks gather an impressive 39.1% of the picks. Consequently, the items that belong to it deserve to be studied in more detail. Beta clustering has also a new category called *advanced visualisation*, which did not include any item as it was added after the reference study was finished. Nevertheless, this category attracted considerable interest during the interview process. Though the questioned sample is considered rather limited, these choices seem to be a clear indication towards prioritising items from certain categories for implementation.

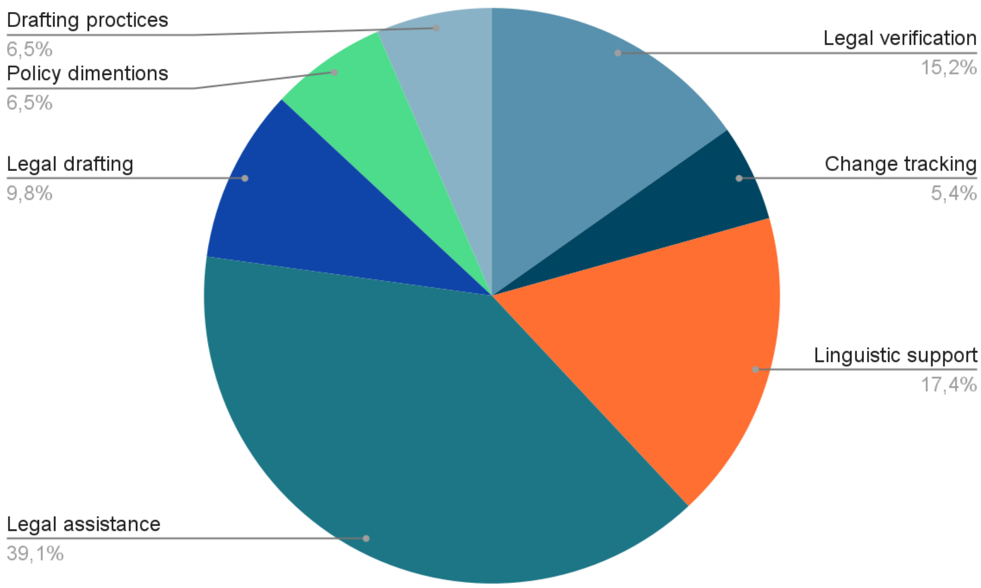


Figure 4.2. Distribution of expert choices (in %) according to beta categorisation (short titles).

It needs to be mentioned that four new or combined smart functionalities were collected during the questionnaire evaluation process, as the EC experts were also asked to provide smart functionalities not present in the original list. *Table 4.2* includes the original suggestion with minor editing for context and enhanced clarity, while also attempting to match these items with specific categories (beta).

#	New smart functionalities	Relates to category(-ies)
1	Detection of contextual incoherences and/or discrepancies within the act: Many errors in acts can be avoided by comparing information within the text (internal references, amounts, spelling of names, word order …)	Verification & linguistics support (related to item #12)
2	Automatic recognition of articles with definitions, hence automatically applying correct article structure. Currently, articles containing definitions have to be identified as such by the drafter to ensure correct numbering	Verification (related to item #5) & legal assistance
3	Automatic validation of all necessary aspects and important topics in drafted legislation	Legal assistance
4	Discovery of potential requirements for IT systems to support the dialogue between law and IT and to help smooth implementation of the law, while supporting the upcoming legal obligation to perform interoperability assessments (see Art. 3 Interoperable Europe Act proposal)	Legal assistance (related to item #19)

Table 4.2. New smart functionalities extracted from questionnaires.

The aforementioned analysis represents an initial, high-level assessment that serves as a foundation for further in-depth exploration. Subsequent stages of the analysis will involve a more detailed scrutiny of the collected data, allowing for a comprehensive understanding of the underlying patterns and trends (see *Section 4.3*). Furthermore, though the primary objective of this stage of the study was not to search for new smart functionalities, many useful suggestions resulted from the consultations, mainly from the interviews as reported in the next section - an interesting and welcome by-product of the study.

4.2.2 Interview results

This section presents the main points from the interviews that are only related to new (or combined) smart functionalities and links them to the different categories (beta). These are manually extracted from the transcripts, edited on a small scale for better understanding, and collected in *Table 4.3*. Already existing smart functionalities are not mentioned. The remarks of the interviewees were adjusted to formulate specific proposals that can be used in the context of the study. Already existing smart functionalities are not mentioned. The table does not include multiple representations of the same proposal. Also, in line with the context of the study, wider proposals not immediately associated with an augmented LEOS were not taken on board.

#	New smart functionalities	Relates to category(-ies)[26]
1	Automatic recognition whether or not a given text belongs to a recital or into the explanatory memorandum	Legal assistance
2	Smart templates - Detection/suggestion of the proper legal template for any legal text	Legal assistance
3	Detection/evaluation of the legal basis (or bases) in view of the content of the document	Legal assistance
4	Coherency check if an act transposed correctly or in line with union, international obligations etc.	Policy dimension
5	Create a smart search facility	Legal assistance
6	Functions allowing to visualise information out of a basic act	Visualisation
7	Automatically draft legal text using imported text from identified data sources	Automatic legal drafting
8	Create and update a database of legal bases	Verification
9	Avoid common errors based on the predictability of drafting customs	Automatic legal drafting
10	Create a table of content of any act	Legal drafting practices (item #32)
11	Summarisation of large legal texts	Policy dimension

[26] Short titles of beta categories.

12	Create and update terminology databases	Verification
13	Automatic e-briefing (and other types of secondary text generation: reporting; fact sheets; Q&As; etc.)	Policy dimension
14	Terminology extraction tool	Legal drafting practices
15	Keeping track of the origin of data	Change tracking
16	Style and quality feature validation based on predefined rules and conventions (joint handbook, inter-institutional style guide, etc.)	Verification
17	Conduct interoperability assessments for digital ready legislation	Legal assistance (see also item #19)
18	Legal processes visualisation & gamification	Visualisation
19	Auto drafting from hints in track-changes and notes from the collaborators	Automatic legal drafting
20	Detect liabilities	Legal assistance (item #19)
21	Maintain style formatting in LEOS/EdiT when importing text (including comments) from external sources	Verification
22	Automatic switch from American to British English	Linguistics support (could be associated with item #11)
23	Presentation of examples, e.g. alternative dispute settlement procedures	Legal assistance
24	Watermark or entirely block printouts to prevent leaking	Policy dimension
25	Filter out/cluster style guide changes and accept them in one batch	Verification
26	Detect deviations in legal jargon and replace this in-house jargon	Linguistics support
27	Use of inclusive language in terms of gender, religion, disabilities, race etc.	Policy dimension

Table 4.3. New smart functionalities extracted from interviews.

Regarding the intervention of LLMs, the table indicates smart functionalities for *summarising*, *briefings*, *reporting* and other drafting needs. In principle, these can all be covered by a single solution rather than several smaller ones. This, however, is linked to implementation aspects that will be studied closer under *Section 5*. In addition, there is evidence of more advanced smart functionalities that resemble entire systems on their own, for instance for conducting *interoperability* or *environmental impact assessments*. These could be also related with item #28: *measure impact of a legislative draft* that belongs to the *policy dimension*.

4.3 Elements of a revised categorisation

The above analysis can be used for investigating the necessity for a revised (gamma) categorisation. As a first step, the distribution of new functionalities that surfaced from the interviews and the questionnaire were depicted in *Figure 4.3*. Each new functionality enters single-weighted in the graph. In case there are two possible categories, the most dominant one is preferred. A total of 44 novel smart functionalities were identified during the interviews. After a thorough process of refinement and cross-checking, this number was narrowed down to 27 distinct functionalities. If added to the original list, the number of smart functionalities increases from 34 to 61. Notably, all four smart functionalities that were initially mentioned in the questionnaires were also reiterated in the interviews. Consequently, these newly discovered items were aligned with their respective categories.

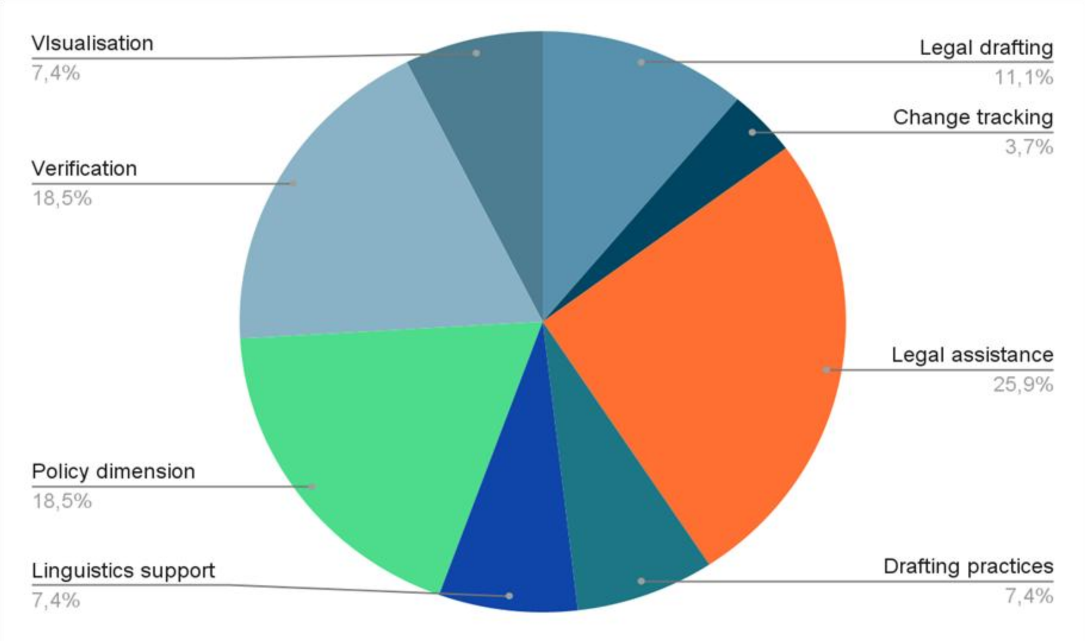


Figure 4.3. Distribution of new functionalities (interviews and questionnaire).

What stands out is that all beta categories are represented in this sample, with the *legal assistance* category being most dominant (7 items or 25.9%) followed by the *policy dimension* and *verification* (each with 5 items or 18.5%). There are overall only a couple functionalities that are related to the visualisation of various types of processes and data. Hence, the project experts find it difficult to consider it a separate category and suggest placing it into the policy dimension. Taking into consideration the above, a *gamma categorisation* is suggested that contains seven categories: that are titled as follows: **verification**; **change tracking**; **linguistics support**; **legal assistance**; **automated drafting**; **legal practices**; and **policy dimension**.

Alpha	Beta	Gamma
Verification I	Context-aware legal verification	Verification
Verification II		
Granular change tracking	Fine granular change tracking	Change tracking
Linguistics support	Linguistics support	Linguistics support
Legal Assistance I	Legal assistance	Legal assistance
Legal Assistance II		
Support 'automatic' legal drafting	'Automatic' legal drafting	Automated drafting
Discovery of Practices	Discovery of legal drafting practices	Legal practices
Policy dimension	Input on policy dimensions	Policy dimensions
	Advanced visualisation	

Figure 5. Categorisation process from alpha to gamma.

The progression of categories from alpha to gamma is illustrated in *Figure 4.4*. Ultimately, gamma categorization proves to be more succinct and less redundant. Reducing redundancy in categorisation efforts contributes to better organisation and enhanced clarity, while improving the consistency and effectiveness of the categorisation system. More specifically, *Figure 4.5* displays the 11 prioritised smart functionalities according to the gamma categorisation. Only four out of the seven categories are represented in the top picks. Legal *assistance* and *linguistic support* stand out among these, with 5 and 4 picks, respectively. *Verification* and *automated drafting* have one mention each.

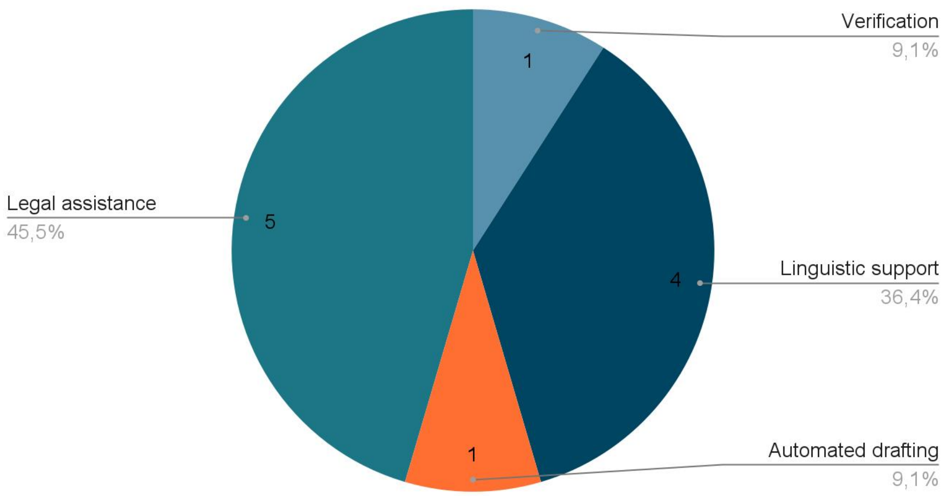


Figure 4.5. Categories of the prioritised 11 smart functionalities.

Moreover, and according to the methodology for implementing the smart functionalities (see *Section 3.3*, a set of attributes can be assigned to each one of these categories, as a first step to approach the business value and techno-business feasibility of the smart functionalities they include. This predefined set of attributes encompasses user experience, potential business value, technology stack, aspects of related data sets, and performance considerations. The high-level description of the defining attributes for each category is shown in *Appendix IV*. The gamma categorisation is used, while the most dominant expressions within each cluster are approximated.

As can be noticed in the above matrix, there are some crucial key aspects that span multiple features and/or system architectural components. Specifically, features that require multiple visual items directly within the workspace of the user need to be implemented in a non-intrusive manner to avoid disruption and additional complexity in the user's work, while at the same time remain easily accessible when needed. On the other hand, due to the processing needs of the necessary algorithms and the multitude and scale of the datasets, it is crucial that mechanisms must be incorporated within the implementation and the hosting infrastructure that can handle the workload. It must be noted that a combination of custom implementation and third party commercial software should be utilised for harnessing the complex specifications of smart functionalities.

5 Technology of Smart Functionalities

This section presents the five main technologies that can be used for the implementation of the prioritised smart functionalities. These are already established and tested technologies, whose underlying algorithms can be embedded in LEOS, for instance through user experience add-ons and extensions and their respective application programming interfaces (APIs), which will provide the backend functionality through a common integration platform.

Due to the high volatility inherent in the AI sector, three indicative open-source solutions for each technology are outlined. It is important to emphasise that attempting a complete end-to-end integration of these tools into LEOS will not be pursued, given the uncertainties in both user experience and technical specifications. Hence, a fundamental approach for integrating into the LEOS system will be outlined. A cohesive and unified integration strategy is proposed, designed to seamlessly incorporate the full spectrum of smart functionalities within the system. This approach ensures that all components work in harmony, optimising the overall efficiency and effectiveness of the LEOS system.

5.1 Advanced Language Editing and Correction (ALEC)

5.1.1 Definition

Advanced Language Editing and Correction (ALEC) refers to the application of computational linguistics and machine learning techniques to detect and correct errors in written text. This field is a subset of Natural Language Processing (NLP), also known as Grammatical Error Correction (GEC). ALEC aims to improve the quality of text by identifying and rectifying grammatical, orthographic, syntactic, punctuation, and stylistic errors (Bryant et al., 2023).

5.1.2 Technology analysis

ALEC was initially based on the application of standard Machine Learning Classifiers. These were originally popular for ALEC, especially for common English as a Second Language (ESL) errors like articles and prepositions. They used various features representing word context and grammar, such as POS tags and dependency relations. Techniques range from simpler models like naive Bayes to advanced neural networks (Han et al., 2006; Lee, 2004; Rozovskaya & Roth, 2011). More recent approaches include:

- Statistical Machine Translation (SMT): SMT approaches ALEC as a translation issue, correcting all error types simultaneously. It uses a noisy channel model, combining a language and a translation model, but can struggle with overall sentence structure and data quality (Brockett et al., 2006; Mizumoto et al., 2011);
- Neural Machine Translation (NMT): NMT uses deep learning for an end-to-end approach in ALEC, employing architectures like RNNs, CNNs, and Transformers. It eliminates the need for complex feature engineering but requires extensive training data (Bahdanau et al., 2015; Vaswani et al., 2017). A prevalent method under this category of approaches is the Seq2seq. Seq2seq text generation can be thought of as a translation engine that translates from a given language to the same language while correcting the grammatical

errors (Yuan & Briscoe, 2016). This approach has been documented to achieve state-of-the-art performance (Vaswani et al., 2017), but it suffers from certain shortcomings, such as long inference and output generation times, the requirement of large amounts of data for training, and the complexity of neural networks;

- Edit-based Approaches: This method generates a sequence of edits rather than complete sentences, improving inference speed. However, it can be limited in addressing multi-token fluency edits (Stahlberg & Kumar, 2020). One important approach in this category is Sequence Tagging (Malmi et al., 2019). It involves tokenizing the incoming text, tagging it, and then mapping it back to corrected tokens. This approach is faster and requires less computational resources compared to neural approaches such as the seq2seq, but it may not be as accurate in correcting complex grammatical errors;
- Language Models: These models use techniques like n-gram or Transformer-based systems for ALEC, which are especially useful in low-resource settings. They rely on large pre-trained models and can correct various error types, though they may overcorrect for fluency (Alikaniotis & Raheja, 2019; Flachs et al., 2019).

5.1.3 State-of-the-art

ALEC's state of the art is represented by Large Language Models such as OpenAI's GPT-3 and GPT-4. These models have demonstrated impressive performance across a variety of tasks, including grammatical error correction (GEC) (Loem et al., 2023). GPT-4, for instance, has achieved a new high score on the JFLEG benchmark, a major GEC benchmark (Coyne et al., 2023). The performance of these models on ALEC tasks is evaluated through human evaluation experiments, where the models' corrections are compared to the source, human reference, and baseline ALEC system sentences.

In the world of commercial systems, Grammarly stands out as one of the leading language editing and correction tools. The details about how Grammarly works are not publicly available because they are confidential. However, it is known that Grammarly relies on a mix of rule-based and statistical machine-learning techniques to identify and fix errors. Some experts believe that Grammarly does not use deep learning algorithms because there is not a sufficient amount of reliable training data available for this type of approach.

5.1.4 Open-source solutions

There are several open-source solutions that integrate ALEC functionality, making them potential candidates for LEOS integration. In this section, three of these solutions are highlighted as particularly interesting for implementation, as assessed by the project experts.

Language:[27] LanguageTool is an open-source proofreading software that supports over 30 languages. It is known for its accuracy and adaptability to various writing styles. LanguageTool offers a user-friendly interface and provides detailed explanations for suggested corrections,

[27] LanguageTool: https://languagetool.org/

helping users improve their writing skills. It can be used via a web interface in a web browser, or via client-side plugins for various applications like Microsoft Office, LibreOffice, Apache OpenOffice, Vim, Emacs, Firefox, Thunderbird, and Google Chrome.

Proselint:[28] Proselint is an open-source library for prose. It provides grammar advice and performs style checks to catch clichés and slang. It is a useful and objective look at prose; one is free to ignore or follow its advice. You can install Proselint as a Python module and run it against a text file.

LM-Critic:[29] LM-Critic is an unsupervised method for grammatical error correction (GEC) that leverages a pretrained language model (LM) to assess the validity of text input. It operates on the principle that a sentence is considered grammatically correct if the LM assigns it a higher probability than its local perturbations. This approach allows for the training of models to fix text errors without needing a perfect critic, which is particularly useful in domains where such a critic does not exist. LM-Critic, along with the Break-It-Fix-It (BIFI) framework, has been shown to outperform existing methods on ALEC benchmarks across multiple domains.

5.2 Named Entity Recognition (NER)

5.2.1 Definition

Named Entity Recognition (NER) is a sub-task of information extraction in Natural Language Processing (NLP) that identifies and classifies named entities in unstructured text into predefined categories (Sharma et al., 2022). These categories can include person names, organisations, locations, medical codes, time expressions, quantities, monetary values, and more. Named entities refer to the key subjects of a piece of text, such as names, locations, companies, events, and products, as well as themes, topics, times, monetary values, and percentages. NER is also referred to as entity extraction, chunking, and identification.

5.2.2 Technology analysis

NER uses algorithms that function based on grammar, statistical NLP models, and predictive models. These algorithms are trained on datasets that people label with predefined named entity categories, such as people, locations, organisations, expressions, percentages, and monetary values.

The most important methods in Named Entity Recognition (NER) can be categorised into three major approaches: rule-based, statistical, and machine learning (Ji et al., 2019)

- Rule-based methods: These methods use a set of handcrafted rules to identify named entities. They often depend on the specific language, domain, and text style, and the compilation process can be time-consuming;

[28] Proselint: https://github.com/amperser/proselint
[29] LM-Critic: https://github.com/michiyasunaga/LM-Critic

- Statistical methods: These methods rely on statistical models trained on annotated data. However, the availability of large-scale general corpora for constructing and evaluating NER systems can be limited;
- Machine learning methods: These methods, including deep learning, use algorithms that learn to identify named entities from large amounts of annotated data.

Deep learning methods, in particular, have shown significant improvements in NER tasks, with transformer-based models like BERT achieving state-of-the-art performance (Li et al., 2022; Yadav & Bethard, 2018).

Hybrid methods have also emerged, intertwining rule-based, statistical, and machine-learning approaches to capture the best of all worlds. These techniques are especially valuable when extracting entities from diverse sources (Bajwa & Kaur, 2015).

5.2.3 State-of-the-art

The state of the art in NER has seen significant advancements in recent years,[30] particularly with the advent of deep learning models. These models have surpassed traditional methods that relied heavily on handcrafted features and domain-specific knowledge.

One of the most successful models in recent years is the ACE+document-context model by Wang et al. (2021), which achieved an F1 score of 94.81 on the CoNLL++ dataset, a cleaner version of the CoNLL 2003 NER task. Other notable models include LUKE (Yamada et al., 2020) with an F1 score of 94.3, CrossWeigh + Flair (Z. Wang et al., 2019) with an F1 score of 93.43, Flair embeddings (Akbik et al., 2018) with an F1 score of 93.89, and BiLSTM-CRF+ELMo (Peters et al., 2018) with an F1 score of 92.22

Deep learning models for NER typically use architectures such as Bi-directional Long Short-Term Memory (BiLSTM), Convolutional Neural Networks (CNN), and Conditional Random Fields (CRF) (Roy, 2021; Yadav & Bethard, 2018). These models are trained on large, annotated corpora and use techniques like BIO notation to differentiate the beginning (B) and the inside (I) of entities, with O used for non-entity tokens. In addition to these models, there are also APIs available for NER tasks, such as those provided by AWS, Google Cloud, IBM, Microsoft Azure, and others. These APIs offer high accuracy, multilingual support, scalability, and ease of integration, making them suitable for processing large amounts of text data in various languages. Despite these advancements, there are still challenges in NER. For instance, state-of-the-art NER models typically report only a single performance measure (F-score), and it's not clear how well they perform for different entity types and genres of text or how robust they are to new, unseen entities (Vajjala & Balasubramaniam, 2022). Furthermore, there is a need for more comprehensive evaluation strategies that take into consideration various text genres and sources, as well as adversarial test sets. In conclusion, the state of the art in NER has seen significant advancements with the advent of deep learning models, but

[30] NER progress tracker: https://nlpprogress.com/english/named_entity_recognition.html

there are still challenges to be addressed, particularly in terms of comprehensive evaluation and robustness to new, unseen entities.

5.2.4 Open-source solutions

Also a popular function, NER can be accommodated via several tools. Three significant open-source ones are presented below.

spaCy:[31] spaCy is a popular open-source library for advanced Natural Language Processing (NLP) in Python. It is designed for production use and is known for its speed and efficiency. spaCy includes a named entity recognition component that uses a convolutional neural network (CNN) and comes with pre-trained models for various languages. It also provides tools for training custom models on your own dataset, making it a versatile choice for NER tasks.

Stanford NER (CoreNLP):[32] Stanford NER, also known as Stanford Named Entity Recogniser, is part of the Stanford CoreNLP suite. It is a Java-based NER tool that uses machine learning to classify entities into predefined categories such as person, organisation, and location. Stanford NER provides pre-trained models for various languages and allows for training on custom datasets. It has been widely used in the research community and is known for its accuracy and robustness.

OpenNLP:[33] Apache OpenNLP is an open-source machine learning toolkit for processing natural language text. It supports various NLP tasks, including tokenization, sentence splitting, part-of-speech tagging, chunking, parsing, and named entity recognition. OpenNLP includes rule-based and statistical NER capabilities and provides a set of pre-trained NER models as well as the ability to train models on custom datasets.

5.3 Semantic Similarity

5.3.1 Definition

Semantic similarity is a metric defined over a set of documents or terms, where the idea of distance between items is based on the likeness of their meaning or semantic content as opposed to lexicographical similarity (Xiong, 2015). These are mathematical tools used to estimate the strength of the semantic relationship between units of language, concepts, or instances, through a numerical description obtained according to the comparison of information supporting their meaning or describing their nature. Semantic similarity is often confused with semantic relatedness (Harispe et al., 2015). Semantic relatedness includes any relation between two terms, while semantic similarity only includes "is a" relations. For example, "car" is similar to "bus", but is also related to "road" and "driving".

[31] spaCy: https://spacy.io/
[32] CoreNLP: https://stanfordnlp.github.io/CoreNLP/ner.html
[33] OpenNLP: https://opennlp.apache.org/

5.3.2 Technology analysis

Semantic similarity can be estimated computationally by defining a topological similarity, using ontologies to define the distance between terms/concepts. For instance, semantic textual similarity (STS) is a key metric used to assess likeness in meaning between terms or documents. It incorporates numerical descriptions that measure the strength of semantic relationships. In other words, semantic similarity is the ability of a computer system to understand the meaning of a piece of text and compare it to another. There are several methods and algorithms used to measure semantic similarity. For example, BERT (Bidirectional Encoder Representations from Transformers), RoBERTa (A Robustly Optimized BERT Pretraining Approach), Sentence-BERT, and ALBERT (A Lite BERT for Self-supervised Learning of Language Representations) are some of the most implemented models for semantic textual similarity (Arase & Tsujii, 2019; Choi et al., 2021; Reimers & Gurevych, 2019; Yang et al., 2020).

5.3.3 State-of-the-art

Semantic similarity plays a crucial role in many tasks such as plagiarism detection, automatic technical survey, semantic search, information retrieval, recommender systems, natural language processing, and more. The state of the art in semantic similarity is constantly evolving with the development of new models and techniques. For instance, a novel method has been proposed for calculating semantic similarity between academic articles using topic events and ontology (Liu et al., 2017). This method represents academic articles with topic events that utilise multiple information profiles, such as research purposes, methodologies, and domains to integrally describe the research work, and calculate the similarity between topic events based on the domain ontology to acquire the semantic similarity between articles. Significant advancement has been achieved also with the use of transfer learning (Jiang et al., 2020) and the evolution of the BERT model with the incorporation of language structures in the pretraining phase (W. Wang et al., 2019).

Moreover, significant advances have been noted in the field of multilingual semantic similarity. The creation of multilingual corpora has been instrumental in advancing the field (Ahmed et al. 2020). These corpora, obtained by crawling bilingual websites or using other data collection methods, provide the necessary data for training and testing multilingual semantic similarity models. Some of the most well-known models in the field are the Universal Sentence Encoder Multilingual module and LASER (Language-Agnostic SEntence Representations), which are designed to handle semantic similarity in a multilingual context (Hirota et al. 2020). These models can process multiple languages and alphabets, making them highly versatile for multilingual semantic similarity tasks.

Semantic similarity is a rapidly evolving field with significant implications for a wide range of applications, particularly in the realm of natural language processing and computational linguistics. The state of the art is characterised by the development and application of increasingly sophisticated models and techniques, driven by advancements in machine learning and Large Language Models.

5.3.4 Open-source solutions

In recent times, text and semantic similarity features have garnered significant attention. As a result, numerous open-source tools are available to achieve the desired functionality. While the ease of integration into LEOS requires individual assessment, the following three particularly interesting tools are briefly introduced.

DKPro Similarity:[34] DKPro Similarity is an open-source framework for text similarity. It provides a comprehensive repository of text similarity measures, including ones based on simple n-grams, stylistic, and phonetic measures. The framework is designed to complement DKPro Core, a collection of software components for natural language processing (NLP) based on the Apache UIMA. DKPro Similarity can be used for a wide variety of tasks, including word choice experiments and Recognising Textual Entailment (RTE) experiments.

ADW (Align, Disambiguate, and Walk):[35] ADW is a software for measuring semantic similarity of arbitrary pairs of lexical items, from word senses to texts. It is based on "Align, Disambiguate, and Walk", a WordNet-based state-of-the-art semantic similarity approach. ADW is available via easy-to-use Java APIs and does not require any training or tuning of parameters.

Semantic Measures Library (SML):[36] The Semantic Measures Library (SML) is a generic and open-source Java library dedicated to the computation and analysis of semantic measures, including semantic similarity, semantic relatedness, and semantic distance. SML also provides a command-line program, SML-Toolkit, which gives access to some of the functionalities of the library, such as computing measure scores. SML can be used to compute semantic similarity and semantic relatedness between entities semantically characterised, such as concepts defined in a taxonomy or entities defined in a semantic graph.

5.4 Natural Language Generation

5.4.1 Definition

Natural Language Generation (NLG) is a subfield of Natural Language Processing (NLP) that focuses on generating coherent and contextually relevant text based on certain inputs. Traditionally, NLG has been considered less challenging than Natural Language Understanding (NLU), but recent advancements, particularly in the area of Large Language Models (LLMs), have significantly revised the NLG research agenda (Dale, 2020).

5.4.2 Technology analysis

The most important Natural Language Generation (NLG) methods can be broadly categorised into traditional rule-based approaches, statistical methods, and deep learning techniques.

---

[34] DKPro Similarity: https://dkpro.github.io/dkpro-similarity/
[35] ADW: https://github.com/pilehvar/ADW
[36] Semantic Measures Library: https://github.com/sharispe/slib

- Rule-based approaches: These methods involve the use of handcrafted rules and templates to generate text. They often require expert knowledge and can be time-consuming to develop and maintain. However, they can produce high-quality output when the domain is well-defined and limited in scope (Santhanam & Shaikh, 2019);
- Statistical methods: Statistical NLG techniques rely on probabilistic models to generate text. These methods often involve n-gram models, Hidden Markov Models (HMMs), or other statistical techniques to predict the next word or phrase in a sequence based on the observed data (Dong et al., 2022);
- Deep learning techniques: With the advent of deep learning, NLG has seen significant advancements. Recurrent Neural Networks (RNNs), Long Short-Term Memory (LSTM) networks, and the Transformer architecture have been instrumental in advancing NLG. These methods have enabled the development of powerful language models, such as OpenAI's GPT-2 and GPT-3, that can generate coherent and fluent text (Lauriola et al., 2022)

In recent years, deep learning techniques based on the transformer architecture with attention mechanism have become the dominant approach in NLG research due to their ability to handle large-scale data and generate high-quality text across various domains.

5.4.3 State-of-the-art

The state of the art in NLG has been significantly influenced by the advent of neural text generation, which has radically revised the NLG research agenda. One of the most visible works in this area is OpenAI’s GPT-2 transformer-based generator. This model and its successors, like GPT-3 and GPT-4, have demonstrated impressive capabilities in generating human-like text (Dale, 2020). GPT models belong to the broader field of Large Language Models (LLMs) that have seen significant advancements in recent years, focusing on improving their capabilities, efficiency, and ethical considerations. Here are some of the most recent advancements:

- Self-Improving Models: One of the most notable advancements is the development of models that can generate their own training data to improve themselves. For instance, Google researchers have built an LLM that can generate a set of questions, provide detailed answers to those questions, filter its own answers for the most high-quality output, and then fine-tune itself on the curated answers (Huang et al., 2022);
- Sparse Expert Models: There is a growing interest in developing massive sparse expert models. These models aim to address the unreliability of current LLMs, which often confidently provide inaccurate information (Fedus et al., 2022);
- Improved Model Architectures: Researchers are actively working on creating more efficient and capable LLM architectures. This includes reducing the computational requirements for training and deployment, while simultaneously increasing their generative and understanding capabilities (Zhao et al., 2023; Zhuang et al., 2023);
- Ethical AI and Bias Mitigation: Addressing ethical concerns and biases in LLMs is a prominent area of research. Efforts are being made to ensure that LLMs are fair,

unbiased, and capable of handling sensitive information responsibly (Gallegos et al., 2023);

- Multimodal Capabilities: Future LLMs are expected to handle multiple types of data, such as text, images, and videos, in a unified manner. This opens up possibilities for more versatile applications in fields like content generation, creative arts, and healthcare (Li, 2023; Meskó, 2023);
- Few-shot and Zero-shot Learning: Advancements in few-shot and zero-shot learning techniques are a priority. This allows LLMs to perform tasks with minimal training data, making them more adaptable to new domains and tasks (Brown et al., 2020; Kojima et al., 2022; Perez et al., 2021);
- Personalisation: LLMs are being developed to provide more personalised interactions, whether in customer service, content recommendation, or other applications. This involves better understanding user preferences and contexts (Chen, 2023; Kang et al., 2023; Salemi et al., 2023);
- Robustness and Security: Ensuring the robustness and security of LLMs against adversarial attacks and other vulnerabilities is a critical area of research. As LLMs are increasingly deployed in real-world applications, they need to be protected from potential threats (Khare et al., 2023; Kim et al., 2023);
- Mitigating Repetitions: Researchers are working on methods to mitigate the issue of repetitive text generation in LLMs. For instance, a method called DITTO has been proposed, where the model learns to penalise probabilities of sentence-level repetitions from synthetic repetitive data (Li et al., 2023; Xu et al., 2022);
- Fact-Checking Capabilities: Future LLMs are being developed with the ability to fact-check themselves. This is a promising approach to mitigate issues of inaccuracy and develop more accurate models (Augenstein et al., 2023; Cao et al., 2023; Lee et al., 2020).

These advancements are expected to significantly enhance the NLG capabilities of LLMs, making them more reliable, efficient, and ethical.

5.4.4 Open-source solutions

In this section, we introduce three widely used open-source solutions that offer capabilities in Natural Language Generation for a LEOS smart functionality. It is important to note that this is just a representative selection, and numerous alternatives are currently available, with even more anticipated in the coming years. Identifying the most suitable candidate for LEOS integration involves tackling a complex scientific and political challenge. Therefore, a thorough investigation is necessary, considering the impending AI Act and other relevant factors.

LLaMA 2:[37] LLaMA 2, an advanced collection of large language models developed by Meta AI, offers models ranging from 7 billion to 70 billion parameters. These models are trained on 2 trillion tokens and have a context window of 4096 tokens. This allows LLaMA 2 to handle more information, beneficial for tasks involving long document understanding, chat histories, and

[37] LLaMA 2: https://ai.meta.com/llama/

summarisation. The models incorporate several architectural improvements, such as the grouped-query attention mechanism, SwiGLU activation functions, rotary positional embeddings, and RMSNorm pre-normalization. LLaMA 2 includes variants like LLaMA Chat, fine-tuned for dialogue use cases, and Code LLaMA, a code generation model. The fine-tuned models have been trained on over 1 million human annotations, and LLaMA 2 has demonstrated superior performance across various external benchmarks. Meta AI has made LLaMA 2 available as an open-access model, enabling unrestricted access to corporations, researchers, and individuals.

BLOOM (BigScience Large Open-science Open-access Multilingual language model):[38] BLOOM was created through the collaboration of more than 1,000 AI researchers at BigScience Research. It was trained between March and July 2022 with about 366 billion tokens. BLOOM stands out with its 176 billion parameters and uses a pure transformer architecture. It presents itself as a compelling alternative to OpenAI's GPT-3

Mistral 7B:[39] Mistral 7B is the first Large Language Model (LLM) developed by Mistral AI, a French AI startup. This model is an open-source Foundation model with 7 billion parameters. It is available for download and use without restrictions, and its raw model weights can be downloaded from the documentation and on Hugging Face. Mistral 7B is a further refinement of other "small" large language models like Llama 2, offering similar capabilities at a considerably smaller size. The model can be used for both research and commercial purposes and is available on GitHub under an Apache 2.0 licence. Mistral AI provides a Docker image bundling vLLM, a fast Python inference server, with everything required to run their model, allowing users to quickly spin a completion API on any major cloud provider with NVIDIA GPUs. Mistral AI's open models aim to offer superior adaptability, enabling customisation to specific tasks and user needs.

5.5 Information Extraction

5.5.1 Definition

Information Extraction (IE) is the process of automatically extracting structured information from unstructured and/or semi-structured machine-readable documents. It involves transforming an unstructured text or a collection of texts into sets of facts that are further processed and stored in a structured format. This process enables the retrieval of specific information related to a selected topic from a body or bodies of text. Information extraction can be applied to a wide range of textual sources, from emails and web pages to reports, presentations, legal documents, and scientific papers (Adnan & Akbar, 2019).

[38] BLOOM: https://bigscience.huggingface.co/blog/bloom
[39] Mistral 7B: https://mistral.ai/news/announcing-mistral-7b/

5.5.2 Technology analysis

There are many NLP technologies that are used in IE in order for the systems to provide accurate and reliable information to the users. Some of the most important are the following:

- Named Entity Recognition (NER): This technique seeks to locate and classify named entities mentioned in unstructured text into predefined categories such as person names, organisations, locations, medical codes, time expressions, quantities, monetary values, percentages, etc. (see *Section 5.2*);
- Sentiment Analysis: This technique is used to identify and extract subjective information from the source material. It helps in determining the attitude, sentiments, evaluations, appraisals, and emotions of a speaker or writer with respect to some topic or the overall contextual polarity of a document (Liu, 2012; Wilson et al., 2009);
- Text Summarization: This technique involves reducing the source text into a shorter version, eliminating the redundant information, and focusing on the main points. It helps in understanding the gist of the text without going into much detail (Sharma & Sharma, 2022). This technique is utilised to identify specific aspects or features of a product or service that customers have expressed their opinions on (Bordoloi & Biswas, 2023; Zhu et al., 2022);
- Knowledge Engineering Approach: This approach involves using domain-specific knowledge and handcrafted rules to identify and extract relevant information from unstructured text. This approach contrasts with machine learning-based methods, which rely on training data and algorithms to learn patterns for extraction. Knowledge engineering methods often involve the use of ontologies, which are formal representations of knowledge within a specific domain, to guide the extraction process (García-Remesal et al., 2010; Vlachidis & Tudhope, 2016);
- Machine Learning Approach: This approach involves using regular expressions along with supervised learning algorithms. The extraction decisions are led by a set of classifiers instead of sophisticated linguistic analyses (Téllez-Valero et al., 2005).

5.5.3 State-of-the-art

The state of the art in information extraction has seen significant advancements, particularly with the application of deep learning techniques. These techniques have been used to extract specific types of entities, relationships, events, and other factual information from unstructured data (Doan et al., 2006; Yang et al., 2022).

Focusing on the state of the art in legal documents information extraction there are a variety of techniques and methodologies, including deep learning, natural language processing (NLP), and ontology-based approaches:

- Deep learning has been used to extract information from legal documents, with advancements in this area being driven by the development of models like BERT and XLNet, which have been adapted for legal texts (Mistica et al., 2020). These models have shown promise in extracting complex data from court judgments, although there is still room for improvement and further research;

- Natural Language Processing (NLP) is another key technique used in legal information extraction. It is used to understand questions and search through information to find relevant answers (Abdallah et al., 2023). NLP-based approaches, deep learning-based approaches, and Knowledge-Based Processing (KBP) based approaches have been used for legal text processing (Zadgaonkar & Agrawal, 2021);
- Ontology-based approaches have also been used to extract specific data from extensive legal documents in text format (Buey et al., 2016). The extraction process is guided by the information stored in a special type of ontology that contains knowledge about the structure of different types of documents, as well as references to pertinent extracting mechanisms.

There are also advancements in the field of legal question answering (QA) systems. These systems are designed to generate answers to questions asked in natural languages and use NLP to understand questions and search through information to find relevant answers (Abdallah et al., 2023). However, despite these advancements, information extraction from legal documents remains a challenging task due to the intricate nature and diverse range of legal document systems. The complexity of legal language, the length of documents, and the need for domain-specific knowledge all contribute to these challenges (Cao et al., 2022; Hong et al., 2021).

In conclusion, while significant progress has been made in the field of legal documents information extraction, there are still many open problems and areas for further research. These include improving the performance of deep learning models on legal texts, developing more effective NLP techniques for understanding and processing legal language, and creating more sophisticated ontology-based systems for extracting information from legal documents.

5.5.4 Open-source solutions

Below, three open-source solutions capable of handling information extraction functionality are presented. Depending on the specific requirements, developers may consider employing one of those.

Haystack:[40] Haystack is an open-source framework that provides tools to build flexible and scalable question-answering (QA) systems. It leverages Transformer-based language models and semantic search to interact with data in a highly intuitive manner. This tool is particularly useful for automating information extraction from large volumes of data, which can significantly boost productivity.

RankQA - Neural Question Answering with Answer Re-Ranking:[41] It is a neural question answering system that introduces a third stage for answer re-ranking to the conventional two-

[40] Haystack: https://haystack.deepset.ai/
[41] RankQA: https://github.com/bernhard2202/rankqa

stage process in neural question answering. The conventional process involves retrieving relevant text passages and then using a neural network for machine comprehension to extract the most likely answer. However, these stages are largely isolated, and information from the two phases is not properly fused. RankQA addresses this by adding a third stage that performs an additional answer re-ranking, leveraging different features directly extracted from the QA pipeline, including retrieval and comprehension features. This approach is lightweight, allowing for efficient estimation, and has shown significant performance improvements over other QA pipelines.

Dolly 2.0:[42] Dolly 2.0 is the first open-source, instruction-following Large Language Model (LLM), fine-tuned on a human-generated instruction dataset licensed for research and commercial use. It can be used for a wide range of applications, including brainstorming, classification, closed QA, generation, information extraction, open QA, and summarisation. Dolly 2.0 is based on EleutherAI's Pythia family of models and has been trained on the Databricks Dolly 15k dataset, with 15,000 prompt/response pairs, developed specifically for matching large language models to instructions.

5.6 Integration technology

5.6.1 Overview

The discussion intentionally adopts a broad perspective on the technology stack and architectural components. This approach is chosen to prevent an overly rigid association with particular technologies or products, which could restrict the design flexibility of the distributed system, not only in its initial implementation but also in its future iterations and additional subsystems. However, adopting a consistent technological framework across the system is important through some key components that can support these functionalities. This consistency is crucial to promote uniform integration practices, ensuring that different parts of the system can work together seamlessly and efficiently.

It is important to note that LEOS is available as an open source solution through code.europa.eu and Joinup. Consequently, the SF and/or its components, along with their respective APIs, could also be presented as independent solutions on these platforms. These could then be easily used by users or integrated with other open source solutions, such as EU BERT,[43] ETAPAS,[44] and eTranslation. The suggested architectural elements and approach embrace this approach, which has been successfully demonstrated in practice in a proof-of-concept displaying high degrees of versatility and adaptability (Leventis et al., 2020, 2021).

5.6.2 Implementation attributes

This part describes the suggested approach to some of the implementation attributes that were already described under *Section 3*, i.e. user experience (UX), data sets (DS) and performance

[42] Dolly 2.0: https://huggingface.co/databricks/dolly-v2-12b
[43] EU BERT: https://joinup.ec.europa.eu/collection/euiba-emerging-tech/solution/eu-bert
[44] ETAPAS: https://joinup.ec.europa.eu/collection/govtechconnect/solution/etapas

(PER). Since LEOS UX layer is a web application, custom views (web pages, panels, controls etc.) may be implemented to incorporate the UX needs of each smart functionality. These views will also contain event handlers that are triggered by user actions. The event handlers, depending on the case (such as a button click, a text selection, etc.), will trigger the required operations behind the scenes, in particular, the LEOS service layer and external API calls.

Based on the results of the interviews, it seems that users are keen on obtaining an MS Word-like user experience to initiate and deploy smart functionalities. In this regard, the design of the **user interface** format is of paramount importance. Hence, in the cases mentioned in the next section, the system could underline (or otherwise highlight) the sections, indicating that further information/actions are available. A side panel would display that information for the respective text. The side panel could be constantly displayed or shown only when the user clicks/selects the text. Document screening could be performed as the users are drafting (automated) or only when they manually initiate the screening of the legal document (manual).

Users should be able to interact with the annotated text, selecting or highlighting sections for further actions. Moreover, advanced customisation is significant. The potential impact of feature customisation on the overall design and functionality of the software is high. Customisation issues deserve, therefore, specific attention during the architectural considerations, as they can significantly impact the adaptability and user-centric features.

Users can be presented in advance with options to choose the annotation style (automated or manual), adjust the system's text identification criteria or manually adding/removing annotations. Users might want to have the option to save, export and/or discuss with other drafters the drafted and/or an alternative text. Other useful key elements of the user interface may include help and tutorials (these can be functionality-specific, uncoupled from the broader LEOS help) and feedback mechanisms (e.g., confirmation messages that the system’s suggestions are useful or not).

The aforementioned external features will provide the necessary data that will then be ultimately presented to the user. Multiple, internal or external, datasets may be accessed, aggregated and transformed into the expected structures before being provided. This is inevitable due to the nature of the functionalities that will require cross-referencing and supplementing between different datasets.

Performance is another critical aspect that needs to be considered due to the potential size of the datasets being accessed and the potential overloading of the systems from concurrent API calls. Retrieving information from the backend is a resource-intensive operation, primarily attributed to the substantial size of these datasets. This could strain the underlying infrastructure's resources and reduce LEOS' responsiveness. One of the ways this can be tackled is by incorporating throttling to keep control of the resources based on certain thresholds. This is to ensure that the system is not overloaded with too many concurrent API calls. The asynchronous messaging mechanisms already in place would also benefit such measures. Further details regarding performance issues will be provided in *Section 6.1.3*.

5.6.3 Services, APIs and security

The LEOS service layer includes various LEOS services as well as integration services. These services may be utilised to integrate with external systems and datasets. Such service operations will be in the form of asynchronous execution of API calls to external backend features.

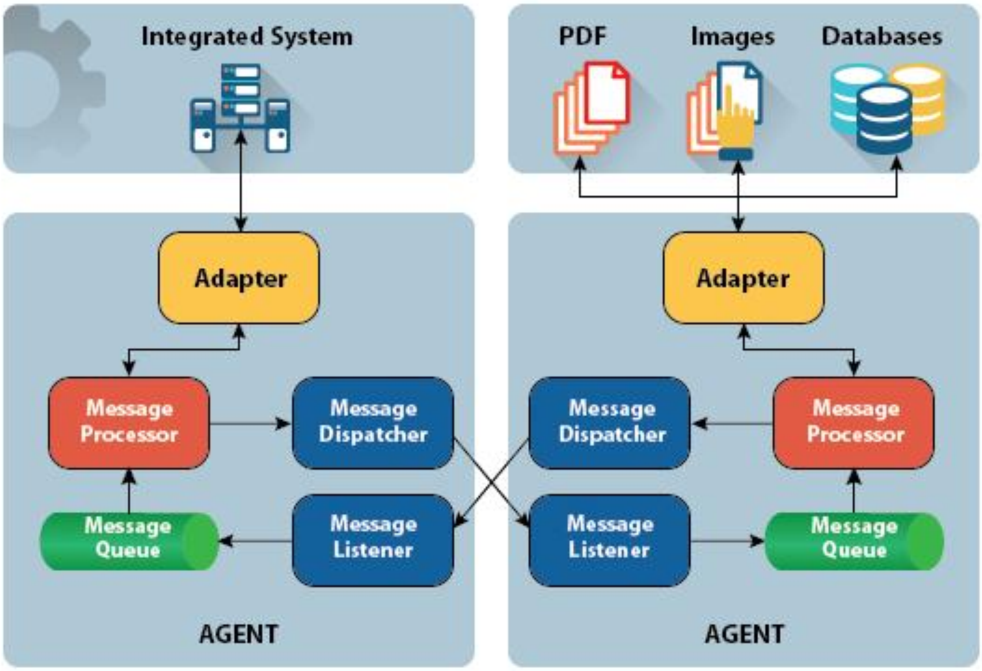


Figure 5.1. Simplified integration architecture.

The APIs will include the software infrastructure needed to support Enterprise Integration Patterns architecture, i.e., design patterns tailored for the development of systems that facilitate the seamless integration of both new and existing software within a business environment (Hohpe & Woolf, 2004). This encompasses message queues that will enable the implementation of a distributed system where each of the subsystems can consume from, and produce data for, the rest of the systems in an asynchronous fashion (Leventis, 2022).

Figure 5.1 exemplifies this approach using agent architecture that enables the integrated (LEOS) system to efficiently employ various datasets in a standardised and consolidated manner. The true benefit emerges when multiple LEOS entities (e.g., from different DGs) require access to these datasets, given that the agent exposing them is already established. This architecture promotes the development of future features, applications, and services offered via an augmented LEOS platform, thereby reducing the time needed to implement production-ready solutions and circumventing repetitive development efforts.

External backend features may include NLG via the inclusion of LLMs, such as ChatGPT, combined with the equivalent UX extensions that can be harnessed during the drafting process.

More specifically, the user may prompt the LLM to respond to questions relating to specific data sets, or the LLM may proactively suggest alternative text or reference data based on the text being drafted. Current hands-on practice in the utilisation of foundation models shows that software integration is likely to involve custom, specially-built prompt engineering behind the scenes.

Last but not least, all backend operations should adhere to the security requirements of the users' organisation and the backend service provider of the external systems. This will entail embedded security elements within the service calls. As might be necessary for specific smart functionality cases, data segments may undergo further processing to remove any user identification in compliance with GDPR and/or be encrypted during transit.

6 Business value and techno-business feasibility of smart functionalities

This section offers a comprehensive analysis of the 11 prioritised SFs within each technology cluster. A thorough examination of the SFs is conducted, elucidating their distinctive attributes. The analysis encompasses their Business Value (BV), and the techno-business feasibility can be deduced by collectively assessing the discussed attributes. These attributes, namely category (CAT), user experience (UX), business value (BV), technology stack (TS), data sets (DS) and performance (PER), are reiterated here for convenient reference.

For the economy of space, it is important to note that attributes with similar characteristics—like TS, DS, and PER— will be discussed in a horizontal manner due to shared commonalities (see *Section 6.1*). While these attributes exhibit such commonalities across many smart functionalities, there are exceptions, as in the case of SF#26 - *LLM-based legal text generation*. Here, the unique attributes will be discussed separately. Overall, the analysis of distinct attributes for each of the said smart functionalities will be conducted in *Section 6.2*, presented in tabular format to improve clarity and consistency.

6.1 Common attributes

6.1.1 Technology stack

From a software development perspective, incorporating any given SF involves embedding of the specific algorithms within a technology cluster. With the major exception of SF#26, a unified approach is suggested to be applied in the development, with adaptations made to align the functionality with UX requirements (see *Section 5.6*). The similarities in the technology stack become apparent in components such as event handling and API calling.

Important considerations include the strategic decision between closed and open-source solutions for incorporating the necessary core functionality. Closed-source solutions provide proprietary, well-supported software with often clear integration paths, e.g., in the form of well-defined APIs. However, such solutions may lack transparency and flexibility, making customisation challenging.

In the case of an open-source approach, pre-existing solutions are used to accelerate development. Such open-source software tools can be cost-effective and customisable. Open-source solutions may offer transparency, flexibility and community collaboration. Their code might be modified to meet specific needs. This approach also taps into the collective expertise of the open-source community. In addition, transparency in source code enhances trust and security. Some well-known challenges of open-source tools include potential lack of support, lack of documentation and, not least, security concerns.

While an alternative exists in developing the core AI algorithm from scratch, caution is advised due to its potential impact on development timelines, which may not align with the project's efficiency goals. The emphasis, therefore, is on optimising development speed without compromising the quality and adaptability of the SF.

6.1.2 Datasets

To ensure the optimal deployment and performance of SFs within an augmented LEOS system, one needs to establish and facilitate seamless access to a robust data stack. Such data stacks may include legal and linguistic resources (see *Section 2.4*, or *Section 2.5* in the case of LLM-based smart functionalities). One needs to investigate whether such lists exist at the EC, DG, or Unit level. It is essential to recognise that drafters and their units or DGs possess significant linguistic resources that have grown to significant volumes over time. Consequently, there may be a need to transform or annotate these resources using a widely accepted standard ontology, such as Eurovoc.[45] The datasets' content can be further enriched by extracting information from diverse sources, including websites, prior references, and local or external repositories of legal data (web crawling). This mixed approach ensures the gradual building of a rich, contextually relevant repository, to enhance accuracy and coherence of the outcomes when using smart functionalities.

In practice, such datasets are likely to be located in multiple, decentralised repos. The LEOS system can access various external data sources using specialised software agents with reusable components (Leventis et al., 2020, 2021). For instance, EUR-Lex hosts COM documents dating back to 1960, while the Archive of European Integration contains numerous older COM documents. To access additional EU official publications, one can explore the Publications Office of the EU or the Archive of European Integration. Legal decisions from the Court of Justice, General Court, and the former Civil Service Tribunal are available on EUR-Lex and the EU court website, Curia. National court decisions on EU law matters can be found on Dec.nat, or the National Case Law collection on EUR-Lex. The JURE database on EUR-Lex compiles cases from national and EU courts related to judicial cooperation in civil and commercial matters. The European Data Portal enhances cross-border data comparability by providing metadata references in a standardised format, utilising RDF technology. It also offers translations of metadata descriptions in all 24 languages using machine-translation technology.

In principle, any data source offering legal information in a structured format can be utilised. However, caution is advised when using alternative or unstructured data, as this may result in lower accuracy during search and processing operations. This underscores the significance of relying on well-organised legal data to enhance the effectiveness of SFs in the legislative process. In the case of translation operations (see, e.g., SF#11), access to datasets containing translation and linguistic reference data, both within and outside EC workspaces. Here, emphasising the utilisation of internal databases is crucial, as drafters tend to prioritise them for their enhanced value compared to external sources. Careful structuring of and linkage to these internal resources ensures not only accessibility but also maximises the SF's efficacy by leveraging the valuable insights and context embedded within the internal translation databases.

One needs to be aware that access to external databases may not always be required. For instance, SF#14 may rely solely on local database operations, since correlations are made within a single act. This applies particularly when the SF demands real-time or context-specific information stored locally, thus optimising data handling efficiency.

[45] EuroVoc: https://data.europa.eu/data/datasets/eurovoc

6.1.3 Performance

For optimal implementation, any smart functionality must exhibit high processing speed, ensuring real-time legal/linguistic analysis. As a general rule, disruption of drafting during data set retrievals needs to be minimised. Ensuring reliable and seamless access to data during dataset retrievals is at the core of performance concerns. In the context of SF#3, for instance, such issues may arise due to the disruption of drafting during the verification process of the correct use of acronyms and other abbreviations.

Consideration of performance is also crucial in the case of accessing linguistic datasets of substantial size because of the risk of overloading the system with concurrent API calls during drafting. This could result in operational disruptions and subsequent pitfalls in system performance, thus impacting LEOS responsiveness. Therefore, managing performance issues during SF operation is crucial to achieve, among others, acceptable levels of UX.

Developers must not only address these aspects but also focus on the smooth implementation of additional back-end operations to meet stringent legal requirements, including compliance with GDPR and cybersecurity measures. To avoid disruptions, developers may choose to implement load-balancing techniques that may include for example asynchronous processing, parallelisation, and caching. These strategies ensure efficient data handling, allowing simultaneous drafting by several users and preventing delays in systems reliant on datasets of substantial volume. These mechanisms play a pivotal role in optimising resource allocation, preventing service interruptions and ensuring a seamless drafting experience within an augmented LEOS (eco)system.

6.2 Analysis of distinct attributes

6.2.1 Cluster I analysis - Advanced language editing and correction

Technology cluster I contains four smart functionalities whose attributes are analysed in *Table 6.1.*

SF	**#9 - Use correct linguistic formulations within the structure of the document**
CAT	Linguistic support
UX	The UX follows the broad approach described in *Section 3.3*. Hence, in terms of annotating, the sections will relate to the linguistic formulations of interest, the system may indicate that alternate linguistic formulations exist by underlining (or otherwise highlighting) those sections. The side panel will display the alternate formulations of the respective text.
BV	While the linguistic formulations mentioned may not directly reference legal texts, it's reasonable to infer that the majority of documents produced within LEOS will be of a legislative nature. Given that EC documents are typically formal, structured, and legal in nature, they can sometimes include vague or ambiguous elements, which might lead to uncertainty in how they're interpreted (Gotti, 2014). Therefore, a smart functionality within LEOS that evaluates the linguistic correctness of these documents and suggests alternative formulations could be extremely valuable. Legal drafters often develop their own unique styles or linguistic structures over time, so

	introducing a software service that compares these individual styles to standard, proven forms or formulations is crucial. Such a system would help create more coherent, high-quality, standardised, and less ambiguous legal documents, reducing the likelihood of these documents being challenged in courts. In turn, this could lower the overall regulatory burden caused by expensive and time-consuming judicial challenges. It is important to note that the concept of 'correctness' in this context is not absolute. Rather, it refers to formulations that are linguistically aligned with a set of legal provisions that are semantically similar to the document being drafted.
SF	**#10 - Correct formulation in accordance with the English Style Guide (ESG)**
CAT	Linguistic support
UX	The user experience for this SF can be characterised in a manner analogous to the previously mentioned one. Essentially, this SF can serve as a sub-part of the former. The difference stems from the fact that the (textual) point of reference is defined in much more narrow terms, the ESG. Similarly, in the user interface case, annotation is unnecessary to display any differentiation from the previous approach. The annotation style can follow similar lines (automated or manual). In the present case, “correct” seems to have the meaning of “ESG-compliant.” As the drafter works on the text, this can be processed in the background and compared with parts of the ESG, in order to identify similar linguistic patterns or formulations based on predefined criteria. The success rate in correctly identifying matching ESG formulations and the degree of their incorporation in the text under development may form suitable benchmarks for assessing the usefulness of this smart functionality.
BV	The introduction of this specific SF to enhance compliance with the ESG holds considerable BV. The EC experts highlighted the importance of ensuring that the texts align with the house style and clear writing principles. The latter may include but is not limited to the ESG. The use of accurate linguistic expressions not only improves the correctness and precision of generated texts but also contributes to heightened readability and enhanced comprehension by third parties (citizens, stakeholders, courts, etc.). In effect, accurate linguistic formulations can reduce the likelihood of ambiguities or misinterpretations. This, in turn, helps mitigate legal risks associated with unclear language or non-compliance. Developing ESG-compliant texts at the level of DGs may also decrease the likelihood of linguistic improvements in any of the subsequent levels of the legislative process, thus saving time and effort. This smart functionality can potentially streamline the drafting process by (semi-)automatically identifying and correcting linguistic errors. This efficiency leads to time and cost savings in the creation of legal documents. Moreover, ensuring alignment with the ESG aids in maintaining consistency and conformity with linguistic standards that are, nonetheless, evolving constantly. This is particularly crucial for regulatory compliance and legal requirements. In addition, legal documents that comply with language guidelines tend to be more user-friendly. This is particularly important for internal and external stakeholders, including citizens who may need to understand complex legal information. In conclusion, the BV of a SF promoting correct formulation of legal provisions in accordance with the ESG extends beyond mere linguistic precision and encompasses risk mitigation, efficiency gains, improved communication and the cultivation of a positive business image.
SF	**#12 - Suggest linguistic formulations in provisions**
CAT	Linguistic support
UX	The intended UX can be implemented based on established UX practices found in major

	portals that embed such recommendation features. By highlighting the specific linguistic formulation of interest within any given provision, the regulatory initiative is handed over to the user, who can then choose to accept or disregard the system's suggestion. A side panel indicator can be employed to present the system's suggestions to the user as soon as they become available. It is highly recommended to include references to the suggested linguistic formulations, incorporating hyperlinks to analogous EC documents. Users should have the capability, upon request, to seamlessly cite these suggestions within the text or include them as footnotes. (see, e.g., Ref2Link).[46]
BV	This is a specialised version of SF#9, only dedicated to legal provisions. The BV of this SF is deemed exceptionally high, given its ability to enhance the drafting process for legal provisions. By suggesting alternative linguistic formulations, the SF empowers the drafter/reviser to customise and adopt proven examples or best practices that align with the specific circumstances of the case. This not only streamlines the drafting workflow but also ensures a heightened level of consistency across similar provisions. This, again, helps maintain a standardised and coherent legal framework, reducing the risk of inconsistencies or ambiguities within legal documents. By providing intelligent suggestions, this SF advances the drafting and revision process, reducing the time and effort required for EC experts and legal professionals to create or modify specific provisions. This heightened efficiency directly translates to increased productivity within legal teams. It should be noted that this SF can contribute to the gradual building of a repository of legal knowledge by storing and suggesting formulations that have proven effective in similar contexts. This knowledge-sharing component holds the potential to enhance drafting skills. Additionally, it can facilitate the seamless transfer of institutional knowledge within legal teams. Furthermore, as previously emphasised while discussing SF#10, this SF has the potential to help mitigate risks associated with legal challenges through the promotion of established linguistic formulations.
SF	**#13 - Detect and avoid structures that could create issues in legal interpretation**
CAT	Legal assistance
UX	The UX for this SF has a high degree similarity with the precious one (#12). However, it is essential to note that, here, the required functionality is exactly the opposite. Instead of suggesting similar or alternative linguistic structures to achieve the desired legal effect, it is designed to indicate that a certain provision is problematic in the given context. An active notification in the side panel should be enough to increase instant awareness to the user/drafter that the legal structure just composed might need to be reconsidered. A colour code could be useful to indicate the severity of the legal issue, while hyperlinks to the relevant texts should be provided. Such notifications though might be considered intrusive and disturb the drafting process. Hence, as discussed in #9, users can and should be presented in advance with options to choose the desirable systemic interventions, i.e. automated or manual.
BV	The envisioned SF, as recommended by EC experts, serves a crucial role in identifying and preventing potential issues in legal interpretation by examining the historical usage of specific terms and structures within legal acts. This can be achieved by exploiting the differences that exist in a defined legal corpus. Its primary objective is to offer insights into whether and how certain terms or structures have been employed in previous legal acts, thereby assisting drafters in avoiding unclear drafting resulting from the misuse of established legal patterns. To achieve this, the SF necessarily taps into a repository of multi-level jurisprudence encompassing both the European Union (EU) and, at a later stage, individual Member States

[46] Ref2Link: https://joinup.ec.europa.eu/interoperable-europe/ref2link

	(MS). By leveraging this extensive legal knowledge base, the tool provides drafters with a contextual understanding of how particular terms or structures have been interpreted and employed across various legal contexts. By incorporating insights from existing jurisprudence, the smart functionality empowers drafters to make informed decisions about the language and structures they employ in legal acts. This, in turn, contributes to the creation of more precise, consistent, and legally sound documents.
TS	The TS for this SF is described under *Section 6.1.1*. Thus, components such as event handling, and API calling will also be present here. The APIs that will be used will be specific to the detection of divergences between different linguistic translations. The solution for implementing the SF involves semantic similarity algorithms (*Section 5.3*). In addition to the technologies mentioned under *Section 5.3.4*, it might also be interesting to investigate EC's eTranslation service.[47] For example, eTranslation as a machine translation tool, utilises Intellectual Property (IP) and case law domain-specific engines, which are relevant for implementing the described functionality.

Table 6.1. Cluster I SFs.

Within the present context, SF#9 emerges as the more generic smart functionality ("within the structure of the document"), appearing to integrate the capabilities of the other two, i.e. SF#10 and SF#12. Specifically, SF#10 represents a narrower version of SF#9, addressing linguistic compliance solely with the English Style Guide (ESG), while SF#12 is considered to be specific to legalistic formulations within legal provisions rather than the entire document. These three not only fall under the same category (linguistic support) but also belong to the same technology cluster.

6.2.2 Cluster II analysis - Named Entity Recognition

Technology cluster II contains a single smart functionality that is analysed in *Table 6.2.*

SF	**#3 - (Correct usage of …) Acronyms, organisations and other abbreviations**
CAT	Verification
UX	The system would provide real-time suggestions and auto-complete options as users type acronyms or abbreviations, preventing typos and ensuring consistency in their usage throughout the document. Vice versa, the SF could automatically recognise and format the names of organisations mentioned in the document, while producing the acronyms and abbreviations. More specifically, the system could analyse the context of the document and offer recommendations for the most appropriate usage of acronyms or abbreviations. This includes considerations for industry standards, document type, or specific organisational preferences within a certain unit or DG. This feature could access a database of known acronyms, organisations and other abbreviations or adapt to new entries.[48] All acronyms and abbreviations should be spelled out when first used in the text. The system should allow users to customise the SF based on their specific writing preferences, style guides, and industry or organisation-specific standards. Moreover, it could learn from user corrections and adapt its suggestions over time, thus becoming a personalised and increasingly accurate writing assistant. It should also be possible to create upon demand a table of abbreviations within any

[47] eTranslation: https://commission.europa.eu/resources-partners/etranslation_en
[48] See, for instance, https://www.abbreviations.com/abbr_api.php

	given document.
BV	A SF aimed at supporting the correct usage of acronyms, organisations, and other abbreviations can significantly enhance drafting efficiency by seamlessly incorporating such elements into the legal documents, removing or, at least, restricting the need for manual verification, thus saving valuable time. Such a SF empowers users/drafters to focus on the substance and content of their writing, rather than on technical details that can cause potential errors. In addition, the SF can potentially not only assist in accurate usage of certain terms but also provide quick reference for users who might not be familiar with them. For this, the mentioned database could also contain websites and references. Another benefit of incorporating this SF is that it guarantees uniform formatting of acronyms, organisations, and other abbreviations throughout the document, adhering to a designated style guide or organisational preferences.

Table 6.2. Cluster II SF.

6.2.3 Cluster III analysis - Semantic Similarity

Technology cluster III contains four smart functionalities whose attributes are analysed in *Table 6.3*. It is noted that this cluster contains the most favourable smart functionalities, i.e. SF #14 and #20.

SF	**#11 - Detect divergences between different linguistic translations**
CAT	Linguistic support
UX	The UX of a SF designed to detect divergences between linguistic translations is inherently challenging for achieving optimal user satisfaction. Its successful incorporation needs to ensure a seamless and intuitive interface. The system must display a high degree of customisation, recognising that individual users may have distinct requirements and preferences. This will eventually allow for building a sense of ownership that will ultimately enable the system's endorsement. In practical terms, customisation needs to take place ex ante, possibly via a separate pane. The assessment results of the linguistic translations using semantic similarity technology can be indicated via an automated or manual annotation style (see, e.g., SF #9). An optional comparative visualisation between two (such feature already exists in LEOS) or more linguistic translations of the same test (e.g., in multi-screen modus) can also be considered.
BV	The BV of a SF detecting divergences between linguistic translations is significant, especially in the international and multilingual context of the European Union (EU). This functionality efficiently identifies discrepancies in translations, improving legal precision and promoting coherence across linguistic variations. It guarantees a more accurate representation of legal and regulatory content and reduces the risk of misinterpretation or ambiguity. The reduction in linguistic divergences not only enhances regulatory compliance but also elevates overall legal accuracy. When it comes to cross-border operations, the SF minimises the potential for disputes, errors, and the consequent need for extensive revision, thereby saving valuable time and resources. More specifically, this SF would help both the Translation Service (TS) and the Legal Service (LS) within the EC. While the TS currently employs specialised in-house AI tools, the LS relies on manual efforts to review draft corrigenda and correcting acts. Implementing such a SF could notably decrease the volume of required drafts, consequently reducing the workload on legal experts. Furthermore, it has the potential to minimise litigation stemming from linguistic discrepancies, reducing the need for involvement from national authorities responsible for

	flagging linguistic, terminology, or content disparities.
SF	**#14 - Correlation between recitals and the enacting terms**
CAT	Legal assistance
UX	This SF is thought to accurately link recitals that usually appear at the beginning of a legal document starting with the word "*Whereas*" with their respective enacting terms in the body of the legal text. Within EU law, a recital refers to a written passage that articulates the rationale behind certain provisions of a legal enactment. It achieves this purpose by avoiding formulation in normative language and political argumentation. A typical UX feature for doing so could involve automated highlighting. According to it, the SF could automatically highlight the relevant position(s) of the enacting term(s) when a user clicks on or hovers over a specific recital. This helps in directly linking the introductory context to the corresponding legal obligations. It could also be combined with a cross-referencing mechanism that allows users to click on terms within the recitals, instantly taking them to the relevant section in the enacting terms, and vice versa. An additional visualisation pane could include a dynamically generated side-by-side view of recitals and enacting terms, which would make it easier for drafters and later for any reader to understand the correlations. Certainly, the above can be combined with other features, such as contextual search that not only searches for specific terms but also considers the context provided in the recitals. Of particular significance can be considered the combination of this SF with a user assistance mechanism that provides real-time suggestions to users based on the context of the recitals.
BV	This SF holds significant BV by addressing challenges in legal document comprehension, drafting and review. By seamlessly correlating recitals with enacting terms and vice versa (see relevant UX), the overall drafting efficiency is enhanced, while the working time for sequentially numbering and accurately linking each recital with its respective terms is greatly reduced. The risk of numbering errors is reduced, and the navigation between recitals and terms becomes more user-friendly, thus enhancing overall productivity. Enhancing this SF with user assistance mechanisms for contextual search and linking suggestions could further amplify the system's business impact. This specific feature will also add value to large and complex legal documents, particularly the ones with a high number of recitals, as it promotes readability and comprehension. It also minimises the risk of misinterpretation by providing clear links between recitals and corresponding enacting terms. Thus, for instance, users and drafters can quickly grasp the relationships between explanatory context and specific legal obligations. The frequent absence of such correlation is a repeated remark expressed by the Legal Service. According to the interviews with EC experts, such SF would be useful not only while drafting but also during negotiations.
SF	**# 15 - (Correlation) between previous acts and the new one**
CAT	Legal assistance
UX	During the law-making or amendment process, a dynamic correlation feature that connects the new act with previous acts is deemed crucial. This SF plays a pivotal role in maintaining legal consistency and coherence. Its capacity to intelligently analyse and display pertinent connections ensures a seamless integration of legal language and principles. There are two key timing considerations for leveraging this feature: *ex ante* and *in-process*. The ex-ante approach implemented before the drafters commence drafting, and the in-process approach applied during or possibly after the drafting phase. The ex-ante approach focuses on proactive preparation, providing drafters with comprehensive insights into relevant legal

	frameworks from the outset. The in-process approach allows for real-time correlation, facilitating adjustments and refinements as the drafting unfolds. By offering these dual approaches, the dynamic correlation functionality enhances the overall efficiency and effectiveness of the legal drafting or amending process. A dedicated pane within LEOS could facilitate the visualisation of correlations. Given that multiple correlations may be present in the ongoing legal document, seamless navigation between past acts and the new one is essential. Like any robust SF, customisation plays a pivotal role. Users should have the flexibility to specify the document type (decisions, directives, resolutions, proposals, etc.) or jurisdiction (national, EU, international/universal, etc.) they wish to display or consider. In addition, in-text citation and referencing of associated legal documents are paramount. For instance, adopting a methodology similar to Ref2Link ensures a similar level of functionality. This customisation not only tailors the user experience but also aligns the SF precisely with the user's requirements, thereby enhancing the overall efficiency and user-friendliness of the legal drafting process within LEOS. It needs to be mentioned that the current SF#15 shares substantial similarities with the upcoming one. Therefore, the prioritisation of both does not come as a surprise. The primary distinction seems to be in the level of automation. In the current one, users initiate the investigation manually, whereas in the SF#20, the process is automated, potentially reducing user involvement in the initial stages of drafting.
BV	Based on the analysis of the interviews conducted with EC experts, it is imperative to ensure coherence when drafting or amending acts. Hence, the incorporation in LEOS of a SF that establishes correlations between previous acts and the new legislation owns substantial BV. Firstly, this feature may enhance the efficiency of the drafting process. Drafters can swiftly identify the relevant previous acts, reducing the time and resources required for research. This time-saving aspect directly translates into increased productivity. Adhering to existing legal language and terminology in new legislation is also vital for clarity, consistency, and legal precedent. It ensures legal precision and minimises ambiguity, leading to effective compliance with established legal norms. Moreover, the risk of errors and inconsistencies in the new legislation is greatly reduced, if not eliminated. By ensuring alignment with existing legislation, the potential for legal disputes or challenges is limited, thus reducing systemic regulatory cost. As a result, this SF empowers drafters to produce higher quality drafts that are aligned with established legal frameworks.
SF	**# 20 - Automatically identify existing legislation relevant for the act under development**
CAT	Legal assistance
UX	The previous SF#15 involves manual initiation of investigations for correlating the new act with previous ones. This one, however, provides increased automation of the identification of pertinent existing legislation for the act under development, with the goal to minimise user intervention, thus streamlining the drafting process. In principle, this option (manual/automated) can be decided in the preparatory phase of drafting any legal document in LEOS and become a distinct customisation feature. The rest of the UX specifications from SF#15 could be transferred to this one too. Naturally, the UX of the SF has to be optimised for drafting efficiency. This is why it needs to involve a seamless, user-friendly interface. A dedicated pane can be used for the presentation of the results of autonomous system scans in legal databases. The presentation of the results has to prioritise clarity by displaying the identified legislation in an organised format that allows users to easily comprehend and integrate the information. Further customisation should allow users to filter results based on specific criteria, such as jurisdiction or legal category.
BV	The analysis of interviews highlights a clear demand among policy officers for a SF that automatically identifies relevant existing legislation during the drafting process. EC experts underscored this as a critical requirement. Notably, a similar EU solution called SeTA

	developed by the Joint Research Centre already employs semantic similarity, aligning with the proposed technology (Hradec et al., 2019). Such a tool would empower authors by providing a comprehensive view of all relevant existing acts, aiding in identifying precedents, evaluating active legislation and ensuring coherence with parliamentary resolutions. This SF not only meets immediate needs but also guarantees consistency between new legislation and the existing acquis, enhancing overall legislative quality and alignment. The resulting rise in productivity and the reduction of potential legal issues can contribute to the overall effectiveness of legislative processes.

Table 6.3. Cluster III SF.

One may observe that SF#15 and SF#20 share numerous similarities. Both fall under the category of *Legal Assistance* and the technology cluster of *Semantic Similarity* (refer to *Section 5.3*). Considering their implementation, it is plausible to merge them into a single SF, as their sole distinction appears to be the level of automation in initiating the relevant feature.

6.2.4 Cluster IV analysis - Natural language generation

Technology cluster IV contains a single smart functionality whose attributes are analysed in *Table 6.4.*

SF	**#26 - Large Language Model (LLM) based legal text generation**
CAT	Automated drafting
UX	Generation of legal text via LLMs is likely to be indispensable in specific facets of law-making. The expected user experience for a SF utilising any LLM (open or closed source) for legal text generation should provide a seamless, efficient and user-friendly interface for anyone involved in legal document creation. ChatGPT by OpenAI is likely to be the point of reference for implementing and operating LLMs in practice for the near future.[49] Similarly any such functionality in LEOS should provide an intuitive interface for users, allowing natural language input and ample customisation options. Moreover, it should offer real-time feedback, ensuring legal accuracy and compliance. Explanatory outputs such as referencing to the actual sources of information via hyperlinks and error handling are expected to enhance transparency and correctness, while substantially reducing hallucinations. The system should possess a learning capability that adapts to user interactions, thus continually improving performance.
BV	The BV of LLMs drafting legal texts can become visible through a couple of examples. Out of the 10k+ EC decisions per year, a significant part concerns repetitive, routine and administrative management decisions. Automated legal drafting could be employed to swiftly generate highly relevant texts. When appropriately supervised by humans, this approach becomes a valuable efficiency enhancement for policy makers. Members of Parliament (MPs) may rely on ChatGPT or any other LLM adopted not only for the composition of parliamentary greetings and speeches but also as a fundamental tool in the initial stages of drafting laws. It is important to highlight that, as discussed by Maruri (2023), significant revisions to the AI-generated content are frequently required to align with specific legislative nuances and structural, linguistic, or even procedural requirements. To date, there are limited investigations on the use of generative AI models in parliaments (Harris and Wilson, 2024). From the above, it can be derived that LLMs, once utilised within a well-defined framework

[49] OpenAI: https://chat.openai.com/

	(see, e.g., Fitsilis et al., 2023), may offer substantial BV in legal processes. For once, they can streamline legal research and analysis by quickly extracting relevant information from vast legal corpora, case law, and scholarly articles, aiding in the identification of pertinent legal principles and arguments. It should also be noted that these models can provide highly specific drafting assistance, generating suggested language aligned with predefined criteria for legislation. This can prove particularly beneficial in complex legal domains where precision is crucial. Additionally, they can contribute to automated compliance checking, comparing drafted legislation with legal requirements to identify potential violations. Moreover, LLMs can be integrated into natural language interfaces or chatbots, enhancing third-party interaction with legal drafts, e.g., by answering questions or explaining concepts within a certain DG or across several ones. Furthermore, they are able to serve in quality assurance and editing of legislative drafts by identifying inconsistencies, grammatical errors and suggesting improvements for clarity and readability.[50] It's emphasised, however, that while these models offer valuable assistance, human judgement and expertise remain indispensable in reviewing the output to ensure accuracy and appropriateness in the legal context.
TS	External backend features of SFs may include a range of NLG-based functionalities achieved through the incorporation of LLMs. This integration can be realised through both on-premise and off-premise LLM deployments. On-premise LLMs offer enhanced data security and control as they operate within the organisation's local infrastructure. On the other hand, off-premise LLMs are hosted on external servers, often provided by trusted vendors through secure APIs. These remote LLMs can be either open or closed systems, with open LLMs allowing broader access and integration capabilities, while closed LLMs offer more restricted, controlled environments for specialised applications. *Section 5.4* describes UX extensions that complement the backend integration. In practical terms, users can leverage these features throughout the drafting process, prompting the LLM—whether on-premise or off-premise—to provide responses to queries related to specific data sets. Additionally, the LLM may proactively suggest alternative text or reference data based on the content being drafted, utilising its NLG capabilities effectively in both open and closed system environments. Hands-on experience of project experts with the integration of foundation models in similar solutions indicates that effective software integration often requires the implementation of custom, specially-built, prompt engineering behind the scenes. This strategic approach ensures a seamless UX while harnessing the full potential of NLG during the drafting process, regardless of the LLM's deployment method or system type.
DS	In the evolving landscape of legal technology, the deployment of LLMs for legal text generation represents a significant leap forward. This process, however, is intricately tied to the nature and quality of the data sets used, the structure and annotation of data, and the tools and resources available to legal professionals. Each data type plays a unique role: structured data helps in understanding formal relationships, unstructured data offers a broader context, and annotated data provides nuanced insights, especially in complex legal scenarios. Understanding the kind and form of data sets suitable for deploying LLMs, the necessity for structured, unstructured, and annotated data, the use of smart templates, and the importance of having access to repositories of prebuilt templates and amendments are crucial components in harnessing the full potential of LLMs in legal drafting. These elements collectively contribute to creating a more efficient, accurate, and compliant legal drafting process. More specifically: *Kind/Form of Data Sets for Deploying LLMs*: For deploying LLMs in legal text generation, both structured and unstructured data sets are relevant. Structured data might include databases of legal decisions, statutes, or regulations, where information is organised in a predictable, field-based manner. Unstructured data can encompass case law, scholarly articles, and other legal

[50] In this sense, LLMs is thought to be possible to facilitate the functionality of the majority of the prioritised solutions in this study.

	texts where information is presented in a free-form, textual manner. Both types are crucial as they provide the foundational knowledge and context LLMs need for generating accurate and relevant legal text. *Necessity for Structured/Unstructured, Annotated Data:* Annotated data is particularly important in the legal domain. For structured data, annotations might include metadata about the legal decisions or statutes, like jurisdiction, relevant legal principles, or case outcomes. For unstructured data, annotations could involve tagging specific legal arguments, identifying precedent-setting passages, or noting critical analyses in scholarly articles. This annotated data helps in training the LLMs to understand the context and nuances of legal language, improving their accuracy and relevance in legal text generation. *Use of Smart Templates:* Smart templates can be extremely beneficial in legal text generation using LLMs. These templates provide a structured framework for generating legal documents, ensuring that the output adheres to specific legal formats and standards. They can also be designed to automatically incorporate user inputs or LLM-generated content in the appropriate sections of the legal document, enhancing efficiency and ensuring consistency. *Access to Repositories of Prebuilt Templates & Amendments:* Access to such repositories is indeed necessary for a comprehensive and efficient legal drafting process using LLMs. These repositories can provide a wide range of templates for different types of legal documents, which can be a significant time-saver. Additionally, having access to amendments or updates to these templates ensures that the documents generated are in line with the latest legal standards and practices. This not only streamlines the drafting process but also helps maintain legal accuracy and compliance.
PER	Performance issues can be mainly attributed to the disruption of drafting during dataset retrieval operations. These might be sufficiently tackled using load-balancing approaches. Though not directly related to this SF, other issues that need to be taken into consideration relate to data privacy and security, which -among others- involve the careful (semi-)automatic assessment of GDPR and cybersecurity guidelines. The need for data privacy and GDPR compliance can impact software system performance due to the potential processing overhead, potentially affecting responsiveness. Other data protection measures, such as encryption and access controls, may introduce additional processing overhead, also affecting system performance. Encryption processes, security checks and real-time monitoring may introduce latency. However, the trade-off is vital to safeguard against potential threats, ensuring the system's robustness and integrity in the face of evolving cybersecurity challenges around LLMs that are currently hotly debated (Gadyatskaya & Papuc, 2023; Naito et al., 2023, Pelrine et al., 2023). Accurate prompting and respective training for experts that utilise LLMs in legal drafting are crucial and can boost performance. On the one hand, clear instructions ensure the generation of precise and legally sound content, enhancing the overall effectiveness of this SF. On the other hand, proper training aids experts in leveraging the technology to its full potential, promoting efficiency and reliability in legal document creation.

Table 6.4. Cluster IV SFs.

6.2.5 Cluster V analysis - Information extraction

Technology cluster V contains a single smart functionality aiming at detecting obligations, rights, permissions, and penalties in legal texts. These are fundamental concepts/elements that define the legal relationships and responsibilities among individuals, entities and the government. In the present SF, four such elements are defined but in the course of its development process, one can also think of incorporating further ones such as: liabilities, entitlements, immunities, remedies, etc. It needs to be underlined that this functionality deals with the detection rather than the generation of such concepts. Its relevant attributes are analysed in *Table 6.5.*

SF	#19 - Detect obligations, rights, permissions, penalties
CAT	Legal assistance
UX	The UX specifications for a SF that detects obligations, rights, permissions, and penalties should be as user-friendly and intuitive as possible. Since the system is meant to detect, visual cues such as highlighting and tagging could help differentiate among different legal elements. Customisation options can ensure adaptability to diverse legal requirements. This could include, for instance, the level of detection, i.e., national (specific MSs), EU-wide, or international. Additionally, the system should provide specific collaboration options (in addition to the one natively provided in LEOS) and legal resources (also in the form of suggestions) in relation to the indicated elements. These specifications aim to create added value when developing such complex legal content under LEOS.
BV	The integration in LEOS of a SF for the detection of obligations, rights, permissions and penalties is a concept that has consistently raised interest among the EC experts, as revealed in multiple interviews in the course of the data collection phase of this project. At the same time, it constitutes an innovative approach that holds the potential to establish a distinct and valuable objective for an augmented LEOS. Indeed, considering the multifaceted nature of law-making, LEOS emerges as an ideal platform for the implementation of this SF. By incorporating the capability to detect (and subsequently manage) the mentioned legal elements, LEOS can significantly enhance its capability in supporting drafters and policy officers to develop legal documents more efficiently. This is because such SF not only simplifies and rationalises the identification of the mentioned legal concepts but also facilitates the rapid detection of potential remedies or penalties, contributing to a more proactive and informed drafting process. Effectively, the law-making process becomes more intelligent and user-friendly, particularly in the context of digital-ready policymaking. For example, the system can identify legal and reporting obligations at the national level and provide suggestions or good practices for addressing them. As certain acts are enacted across EU member states, their coherence can be assessed, ensuring alignment with the drafter's intent. This introduces an additional layer to the processes of transposition, implementation, monitoring and correction. Moreover, this SF holds the potential to significantly assist in verifying whether a draft aligns with the general legal obligations of the European Union, thereby enhancing the overall effectiveness and compliance of legislative initiatives. Overall, this SF would offer considerable gains in legal efficiency and accuracy. Such a system could accelerate the drafting of legal documents, ensuring precision and consistency while minimising the risk of errors associated with inaccurate or wrongful drafting of the mentioned concepts. This again could expedite the drafting process, allowing legal professionals to focus on higher-order tasks, reducing workload and enhancing overall productivity. Additionally, the SF could facilitate real-time updates in response to changes in the legal document, promoting compliance and adaptability within a dynamic legal landscape. Moreover, positioning this advanced functionality within LEOS would serve the operational needs of administrative, executive or law enforcement agencies, while also strategically aligning them with the EC's regulatory objectives.

Table 6.5. Cluster V SFs.

At this point, it needs to be mentioned that in order to enhance regulatory reporting efficiency, the EC, in partnership with the University of Bologna and the University of Liege, has already initiated the Study On Regulatory reporTIng Standards (SORTIS).[51] This pilot project seeks to

[51] SORTIS: https://bit.ly/3HNuUH9

establish a vocabulary for regulatory reporting metadata and create uniform formulations for reporting requirements and associated metadata.

7 High-level roadmap

7.1 General approach

The fourth and final task of the study leads to the development of a high-level roadmap for the implementation and deployment of the smart functionalities that were closely defined in the previous tasks. The roadmap takes into consideration the reference study by the University of Bologna. Moreover, the reference study highlights a set of obstacles that require specific attention and proposes various components of an action plan, serving as valuable inspiration for further development. While the original roadmap serves as an initial guide, it was extended and looked upon from an implementation perspective that incorporates the fresh insights gained during the current study.

It needs to be noted that the Insights from interviews with EC experts indicate organisational readiness and stakeholder support for implementing an augmented LEOS. However, several operational challenges remain uncertain. Therefore, certain aspects of the roadmap aim to address these challenges. As such, the high-level roadmap for implementing smart functionalities into the LEOS system encompasses critical components. These include, among others, ensuring consistency in the technological framework across the EC, supported by key components; establishing seamless access to a robust data stack, including legal and linguistic resources; developing and adapting AI guidelines to fit EC's context; testing of solutions within interoperability regulatory sandboxes. Additionally, attention is needed on aspects such privacy, security, training and aligning with upcoming regulations like the AI Act.

Following these, the roadmap entails developing proofs-of-concept for each functionality cluster, capturing lessons learned, and defining project parameters for agile development of the Augmented LEOS platform incorporating smart functionalities. Throughout the process, it is paramount to maintain open communication channels with stakeholders and continuously monitoring performance for ongoing optimisation. A relevant task force at EC level could be established for this purpose. The different parts of the roadmap and the nature of activities involved are shown in *Table 7.1*. The overall endeavour comprises three essential types of activities: research studies (effort dedicated to research), software development (effort focused on software creation), and consultancy (effort dedicated to advisory services).

Most of these parts constitute ‘enabling factors’ for successfully implementing smart functionalities. These factors were identified during the present study and are considered paramount for integrating smart functionalities into existing or adapted law-making workflows. Also, it seems impractical to assign specific durations or indicative timelines, as these will be depending on factors like budget constraints and the availability of researchers or vendors. The identification of researcher groups for conducting these studies and selecting vendors for developing proofs-of-concept is deemed important at this stage. The parts mentioned above could form components of a comprehensive implementation framework. However, given the innovative and rapidly evolving nature of the task, it is advisable to treat them as separate contracts and engage multiple external research and development stakeholders for each part.

This approach ensures specialised attention and expertise tailored to the unique requirements of each aspect.

Part	Content	Nature
	Preparation	
A	Conduct a comprehensive evaluation of the current EC technology framework to ensure alignment and consistency with the proposed implementation framework	Study
B	Examine privacy and security considerations, including their alignment with the AI Act, as part of the preparatory phase for developing the proposed implementation framework for smart functionalities.	Study
C	Conduct a thorough training needs analysis to identify the specific training requirements and parameters essential for preparing EC stakeholders to effectively utilise the LEOS smart functionalities. It may involve tailored training programs, instructional materials, workshops, and open, online resources.	Study
D	Prepare an in-depth study on LLM dependencies focusing on the issues detailed in *Section 7.3*: foundational models, technical facilitation and training.	Study
E	Customise existing or develop new AI guidelines to align with the specific context of the EC, laying the groundwork for the practical utilisation of the LEOS smart functionalities at a production level.	Study
	Proof-of-Concept	
F	Develop and ensure seamless access to diverse EC data stacks through an augmented LEOS system, with provisions for processing the underlying data sets in accordance with legal document standards.	Software
G	Establish and validate solutions within interoperability regulatory sandboxes to enhance data and system security during the development of augmented LEOS.	Software
H	Develop multiple proofs-of-concept for an augmented LEOS, with each focusing on a distinct smart functionality cluster.	Software
	Follow-up	
I	Document lessons learned and draft tender documents for the development and deployment of production-level augmented LEOS solutions.	Consultancy
J	Promote dissemination and community building thus advancing stakeholder engagement, with a focus on leveraging the existing community within the EU's Joinup platform, while disseminating project updates and outcomes.	Consultancy

Table 7.1. High-level roadmap parts.

Implementation could be planned in three distinct steps: *preparation*, *proof-of-concept* and *follow-up*. The mentioned parts within each step could be developed sequentially or, alternatively, in a quasi-parallel manner depending on the available resources. Regardless, with a reasonable investment of resources and a high level of prioritisation from the EC, full implementation could be anticipated within a couple of years. While the current roadmap can serve as an initial foundation, it is crucial to recognise that it should be adaptable, accommodating changes and iterations throughout the implementation process. Hence, it is likely to require regular reviews and updates, typically on an annual basis, incorporating stakeholder feedback and ensuring alignment with broader strategic objectives.

7.2 Software development

The recommended software development approach involves creating individual proofs-of-concept for each smart functionality cluster. An indicative timeline for implementing a single proof-of-concept is shown in *Figure 7.1*.

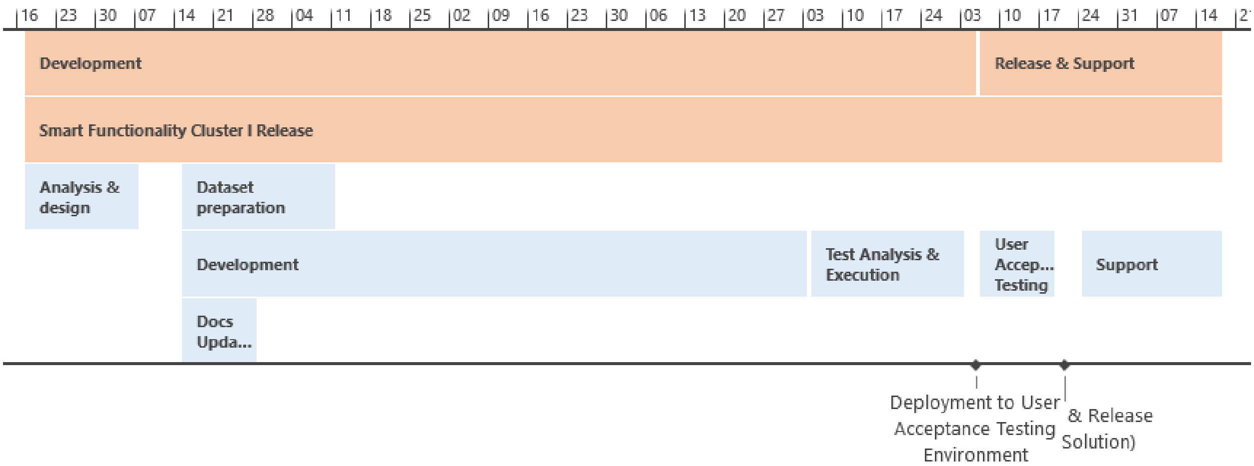


Figure 7.1. Timeline for a single proof-of-concept.

Development and dataset preparations can begin upon confirmation of the project plan for the individual proof-of-concept. As regards the dataset preparation, for each proof-of-concept, it is crucial to evaluate the required data sources and allocate significant resources to develop structured, validated, and open datasets, preferably in a standardised AKN-based format. This approach is likely to be more practical when focusing on specific datasets essential for implementing particular clusters, rather than attempting to create a unified dataset from scratch at the project's outset.

Subsequently, testing and documentation processes may be initiated. Upon successful completion of testing, the developed features are deployed to a User Acceptance Testing (UAT) environment according to specifications and guidelines set for interoperability regulatory sandboxes. This environment is accessible to stakeholders for functionality testing and feedback provision, as illustrated in the project plan's Release and Support phase. After UAT concludes, the developed solution is ready for a (limited) release, such as within a predefined user community. In this scenario, release notes are prepared to accompany the solution's publication, potentially on platforms like Joinup or code.europa.eu.

Overall, proofs-of-concept are possible to be implemented for four out of the five defined clusters using existing software tools, open technologies, and acknowledged development practices. These are technology clusters that have at their core a specific type of algorithms and include **SF cluster I** (*Advanced Language Editing and Correction*), **SF cluster II** (*Named Entity Recognition*), **SF cluster III** (*Semantic Similarity*), and **SF cluster V** (*Information Extraction*). It needs to be highlighted that **SF Cluster IV** (*Natural Language Generation*) is excluded from this approach due its heavy and uncertain dependencies that are outlined further down this chapter.

Following successful implementation, a 'standard' agile approach can then be utilised for the implementation of a production-level augmented LEOS system that incorporates the desired smart functionalities. In this case, the roadmap could be extended with additional part, such as *Project management and governance*. This part will mainly oversee the project's strategic direction, risk management, and resource allocation. It ensures adherence to timelines, specifications, and regulatory requirements, while optimising project outcomes and facilitating effective decision-making throughout the software development cycle. In conclusion, the provided roadmap is seems suitable for implementing the prioritised SFs across four distinct clusters within a timeframe of two years, deliberately excluding the fifth one, "*Natural Language Generation*" for reasons that are presented in the next section.

7.3 Considerations for implementing NLG-based solutions

Given the current landscape surrounding the construction, parameterisation, training, and operation of LLMs, the study considers it necessary to exclude the development of solutions utilising Natural Language Generation (NLG) technology from this roadmap due to three critical dependencies. Without addressing them swiftly and decisively, the relevant smart functionality (#26 - *LLM based legal text generation*) cannot be feasibly implemented. A first approach to these dependencies is attempted:

1. ***Foundational Model Selection***: The selection of foundational models for NLG is a pivotal process, particularly for an organisation like the EC, which has specific operational and regulatory needs. In this regard, the AI Act is expected to add further complexity in the regulatory domain. Model selection involves several considerations:
    a. *Technological Sophistication*: The model must be at the forefront of current AI developments.[52] It should possess advanced capabilities in understanding and generating natural language, ensuring high-quality, contextually appropriate outputs. This requires assessing models based on their architecture, performance metrics, and adaptability to different linguistic contexts and tasks;
    b. *Alignment with Data Sovereignty and Security:* Given the sensitive nature of the data handled by the EC, the chosen model must comply with stringent data protection and privacy regulations, such as the GDPR. This necessitates an on-premises approach where data processing and storage are managed internally, rather than relying on external cloud services. The model must thus be compatible with on-premises deployment, ensuring that all data remains within the EC's control and jurisdiction;
    c. *Complex Selection Process:* The selection process involves rigorous testing, evaluation, and potentially customisation of models to meet specific requirements. This includes assessing the model's ability to handle multiple EU languages and its adaptability to the EC's unique operational context. Considering these factors, it becomes apparent that integrating such a sophisticated model into the EC's systems within the two-year roadmap timeframe is more than challenging.

[52] While singular is used here, it is important to acknowledge the possibility that various models may demonstrate effectiveness across different applications.

2. ***Data Centre Building*****:** Building custom data centres for the training of LLMs is a critical infrastructure requirement due to several factors:
    a. *Computational Demands:* Training LLMs is computationally intensive, requiring powerful processors and substantial memory resources. Custom data centres provide the necessary hardware to handle these demands, including high-performance GPUs and large-scale storage solutions;
    b. *Scalability and Flexibility:* As AI models and algorithms evolve, the data centre(s) must be scalable and flexible to accommodate technological advancements. This means not only having the capacity to expand but also ensuring that the infrastructure can adapt to new computing paradigms and software requirements;
    c. *Energy Efficiency and Sustainability:* Given the high energy consumption of such data centres, designing them with sustainability in mind is crucial. The optimisation of energy usage, while exploring renewable energy sources to minimise the environmental impact is considered the *non plus ultra*;
    d. *Security and Reliability:* For an organisation like the EC, data centre security is paramount. This includes physical security, cybersecurity measures, and robust backup systems to ensure data integrity and continuity of operations.

3. ***Training of LLMs*****:** The process of training LLMs is complex and resource-intensive for several reasons:
    a. *Computational Resources*: LLMs require significant computational power, often necessitating the use of specialised hardware like GPUs or TPUs. This is because the training involves processing vast amounts of data and performing complex calculations;
    b. *Expertise*: Effective training requires a team with expertise in machine learning, natural language processing, and relevant domain knowledge. This expertise is necessary to design, implement, and refine the training process, ensuring that the model learns effectively and accurately. Unfortunately, such expertise is a scarce resource;
    c. *Time Investment*: Training LLMs is not a quick process. It involves iterative cycles of training, testing, and fine-tuning, which can take weeks or even months, depending on the model's complexity, the data volume, and of course the available processing power;
    d. *Data Requirements*: LLMs require large and diverse datasets to learn effectively. Gathering, curating, and preparing these datasets is a significant undertaking. The data must be representative, unbiased, and sufficiently varied to ensure the model can generalise well across different contexts (see also *Section 2.5*);
    e. *Quality of Output:* The ultimate goal of training LLMs is to produce models that can generate accurate, fluent, and contextually relevant text. If the training is inadequate, the models may produce outputs that are inaccurate, nonsensical, or biased, limiting their utility and application in real-world scenarios.

Hence, it becomes evident that the lack of any of these dependencies prevents from implementing any LLM-based solution. A more focused examination of these critical parameters can be conducted through a dedicated study.

8 Conclusions and outlook

This report constitutes the main outcome of an eight-month long technical study entitled "Overview of Smart Functionalities in Drafting Legislation in LEOS". It is centred around a prioritised list of 11 smart functionalities (SFs), selected based on the preferences of interviewed EC experts. Efforts concentrated on technology considerations, while providing detailed descriptions of the five main AI technologies applicable to the implementation of these SFs.

Following the categorisation of SFs and the assessment of technologies necessary for their implementation, the discussion shifted to the defining elements of SFs, including user experience, business value, technology stack, data sets and performance. Each of these aspects was examined for every prioritised SF. The business value was primarily derived from interviews with EC experts. Furthermore, practical considerations such as deployment, system integration, dependencies and implementability were discussed and presented with a pragmatic approach that may serve as a basis for further elaboration.

There are several similarities between the smart functionality #15-*Correlation between previous acts* and the new one #20-*Automatically identify existing legislation relevant for the act under development*. Both belong to the same category (*Legal assistance*) and technology cluster (*Semantic similarity,* see Section 4.3). From an implementation point of view, these could be merged into one single smart functionality, since their only difference seems to be the degree of automation in initiating the relevant feature. Nonetheless, there are potential differences in nuance and granularity as #15 appears to explore the detailed wording of directly related acts, while #20 takes a more scoping-oriented approach.

There are also substantial similarities among smart functionality #9-*Use correct linguistic formulations within the structure of the document*, #10-*Correct formulation in accordance with the English Style Guide*, and #12-*Suggest linguistic formulations in provisions*. All three belong to the same category (*Linguistic support*) and technology cluster (*Advanced language editing and correction*, see Section 4.1). In addition, there also exist a negative correlation between #12-*Suggest linguistic formulations in provisions* and #13-*Detect and avoid structures that could create issues in legal interpretation*. Analysis indicates that a single software implementation can efficiently cover this entire collection of features.

One of the main goals of the study was to investigate the integration of the discussed functionalities within the LEOS system. For this, reliance on proprietary technologies should be reduced, opting instead for an open-source approach. The use of existing EC-funded toolsets, such as the ones that are featured on *Joinup*, could be used to expedite development and seamlessly integrate with legacy systems. This strategic reuse holds the promise of substantially reducing development timelines and facilitating seamless integration with existing legacy systems. Nonetheless, the practicality and cost-effectiveness of integrating such technologies into LEOS depend on the timing of implementation and the specific approaches selected, e.g., private vs. public cloud, LLM model selection, building of data set(s) and training, and reskilling on EC level.

To mitigate potential challenges arising from evolving technologies and changing requirements, it is advisable to commence with the implementation of a select set of functionalities at the earliest opportunity to allow for the early testing and deployment of in-demand features. Such a strategy establishes a valuable feedback loop, offering critical insights and guiding further developments.

The AI technologies discussed in the report have reached a significant level of maturity, reflecting their readiness for implementation under LEOS. However, the dynamic nature of AI and technology suggests that by the time the development begins, the technological landscape might have evolved further. This fast-paced evolution highlights the importance of an adaptable and forward-looking approach in the development and implementation phases. It is crucial to design the system with the flexibility to integrate future advancements seamlessly. To prepare for this, it appears advantageous to initially assess multiple LEOS add-ons incorporating smart functionalities using a proof-of-concept approach. Upon confirming the viability of these solutions, transitioning to a standard agile development approach for their productive implementation would be recommended.

As some of the aforementioned smart functionalities might undergo development in the near future, one might have to consider provisions outlined in the forthcoming Interoperable Europe Act. Attention should be directed towards aspects such as interoperability regulatory sandboxes for the development and testing of innovative solutions (for a more detailed elaboration see OECD, 2023). Additionally, it is essential to anticipate the transition of Joinup into the Interoperable Europe Portal in 2024, as outlined in the aforementioned Act. Notably, LEOS is already established as a solution on Joinup, underscoring its alignment with advancing interoperability initiatives and the relevant regulatory framework.

Inevitably, the report touched on the critical role of open systems and foundational models in the implementation of smart functionalities. The use of open systems aligns with the interoperability requirements and the evolving standards in the tech industry. Foundational models, particularly in the realm of LLMs, play a pivotal role in enhancing the functionalities of LEOS. These models, when integrated effectively, could potentially streamline legal processes, improve accuracy and ensure compliance with legal standards such as *Akoma Ntoso* (AKN) based schemes or, specifically, AKN4EU, which is natively used in LEOS. The latter is made possible due to the ability of LLMs to understand and generate code. However, the implementation of these models must be carefully managed to align with diverse legal-technical constraints.

The algorithms that underpin the smart functionalities of LEOS are advanced and introduce a certain level of complexity. Despite this fact, there is a notable diversity in the open-source systems available, as detailed in *Section 5* of the report. This diversity is considered a strength, offering a range of options for customising and scaling the functionalities according to specific needs. Furthermore, the open-source nature of these systems enhances transparency and trust, crucial for legal and regulatory environments. However, it also brings challenges such as the need for comprehensive support, robust documentation and addressing security concerns.

It is important to highlight that the development of end-user training courses for utilising smart features in LEOS is an essential component of any comprehensive approach towards achieving an Augmented LEOS. These training courses could be designed as eLearning modules and made accessible through platforms such as the EU Academy[53] or the Interoperable Europe Academy project[54]. The report briefly touched upon specific training aspects, notably emphasising the importance of comprehensive training protocols tailored specifically for LLM prompting. Given the rapid advancements in language models, such training is imperative to ensure the effective and responsible utilisation of LLM-based apps.

Several other issues enter the discussion concerning the practical implementation of smart functionalities, including considerations of privacy implications within the context of the General Data Protection Regulation (GDPR, Regulation (EU) 2016/679) and the utilisation of foundation models within the context of the AI Act, projected to be effective in 2026. Despite their critical significance, these legal-technical matters are tied to specific implementation aspects and, therefore, intentionally excluded from the scope of the present report.

It is also advised to prioritise proofs-of-concept for selected smart functionalities with a focus on application at the EC level. This recommendation is underscored by significant disparities observed in the legal and judicial datasets across EU member states, mainly due to the lack of standardisation in such decentralised datasets. In addressing this challenge, the adoption of legal document standards such as AKN could offer long-term benefits by facilitating harmonisation and interoperability among multiple European legal orders.

The high-level roadmap in *Section 7* consists of a number of parts (studies, software and consultancies) for the development and deployment of a set of proofs-of-concept for SFs that were defined in this report. From an implementation perspective, the definition of the attributes that were presented in *Section 6* for each of the SFs serves as the basis on which the technical analysis of the functional and non-functional requirements may be formulated.

The specific design of the technology stack can then be established and, subsequently, the first versions of production-ready SFs may be developed. The exact development plan of each SF and their versions will also depend on organisational priorities and external dependencies along with specific milestones and horizontal activities. Nonetheless, significant considerations emerge regarding the implementation of NLG-based solutions, particularly given the current landscape surrounding LLMs. Therefore, the study opted to exclude the development of such solutions from this roadmap due to critical dependencies. At the same time, a key finding of the study reveals that the majority of indicated smart functionalities does not necessarily require the development or integration of LLMs for achieving essential functionality. Instead, existing open-source technologies and modules can be utilised, mitigating the aforementioned dependencies associated with LLMs.

[53] EU Academy: https://academy.europa.eu/
[54] Interoperable Europe Academy: https://bit.ly/49savDx

9 References

Abdallah, A., Piryani, B., & Jatowt, A. (2023). Exploring the state of the art in legal QA systems. *Journal of Big Data*, *10*(1), 127. https://doi.org/10.1186/s40537-023-00802-8

Adnan, K., & Akbar, R. (2019). An analytical study of information extraction from unstructured and multidimensional big data. *Journal of Big Data*, *6*(1), 91. https://doi.org/10.1186/s40537-019-0254-8

Agnoloni, T., Bartolini, R., Frontini, F., Montemagni, S., Marchetti, C., Quochi, V., ... & Venturi, G. (2022). Making Italian Parliamentary Records Machine-Actionable: the Construction of the ParlaMint-IT corpus. In *Proceedings of the Workshop ParlaCLARIN III within the 13th Language Resources and Evaluation Conference*, pp. 117–124, Marseille, France. European Language Resources Association. https://aclanthology.org/2022.parlaclarin-1.17

Agnoloni, T., Marchetti, C., Battistoni, R., & Briotti, G. (2022). Clustering Similar Amendments at the Italian Senate. In *Proceedings of the Workshop ParlaCLARIN III within the 13th Language Resources and Evaluation Conference*, pp. 39–46, Marseille, France. European Language Resources Association. https://aclanthology.org/2022.parlaclarin-1.7

Ahmed, M., Dixit, C., Mercer, R. E., Khan, A., Samee, M. R., & Urra, F. (2020). Multilingual Corpus Creation for Multilingual Semantic Similarity Task. In *Proceedings of the Twelfth Language Resources and Evaluation Conference* (pp. 4190-4196). European Language Resources Association. https://aclanthology.org/2020.lrec-1.516

Akbik, A., Blythe, D., & Vollgraf, R. (2018). Contextual String Embeddings for Sequence Labeling. In E. M. Bender, L. Derczynski, & P. Isabelle (Eds.), *Proceedings of the 27th International Conference on Computational Linguistics* (pp. 1638-1649). Association for Computational Linguistics. https://aclanthology.org/C18-1139

Alexopoulos, C., Lachana, Z., Androutsopoulou, A., Diamantopoulou, V., Charalabidis, Y., & Loutsaris, M. A. (2019). How Machine Learning is Changing e-Government. In *Proceedings of the 12th International Conference on Theory and Practice of Electronic Governance* (ICEGOV '19). Association for Computing Machinery, New York, NY, USA, 354–363. https://doi.org/10.1145/3326365.3326412

Alikaniotis, D., & Raheja, V. (2019, August). The Unreasonable Effectiveness of Transformer Language Models in Grammatical Error Correction. In H. Yannakoudakis, E. Kochmar, C. Leacock, N. Madnani, I. Pilán, & T. Zesch, *Proceedings of the Fourteenth Workshop on Innovative Use of NLP for Building Educational Applications*, Florence, Italy.

Arase, Y., & Tsujii, J. i. (2019, November). Transfer Fine-Tuning: A BERT Case Study. In K. Inui, J. Jiang, V. Ng, & X. Wan, *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)* Hong Kong, China.

Augenstein, I., Baldwin, T., Cha, M., Chakraborty, T., Ciampaglia, G. L., Corney, D., DiResta, R., Ferrara, E., Hale, S., & Halevy, A. (2023). Factuality challenges in the era of large language models. *arXiv preprint arXiv:2310.05189*.

Bahdanau, D., Cho, K., & Bengio, Y. (2015). *Neural Machine Translation by Jointly Learning to Align and Translate.* http://arxiv.org/abs/1409.0473

Bajwa, K. S., & Kaur, A. (2015). Hybrid Approach for Named Entity Recognition. *International Journal of Computer Applications*, *118*(1), 36-41. https://doi.org/10.5120/20713-3048

Bordoloi, M., & Biswas, S. K. (2023). Sentiment analysis: A survey on design framework, applications and future scopes. *Artif Intell Rev*, 1-56. https://doi.org/10.1007/s10462-023-10442-2

Brockett, C., Dolan, W. B., & Gamon, M. (2006, July). Correcting ESL Errors Using Phrasal SMT Techniques. In N. Calzolari, C. Cardie, & P. Isabelle, *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics* Sydney, Australia.

Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., & Askell, A. (2020). Language models are few-shot learners. *Advances in neural information processing systems*, *33*, 1877-1901.

Bryant, C., Yuan, Z., Qorib, M. R., Cao, H., Ng, H. T., & Briscoe, T. (2023). Grammatical Error Correction: A Survey of the State of the Art. *Computational Linguistics*, *49*(3), 643-701. https://doi.org/10.1162/coli_a_00478

Buey, M. G., Garrido, A. L., Bobed, C., & Ilarri, S. (2016). The AIS Project: Boosting Information Extraction from Legal Documents by using Ontologies. In *International Conference on Agents and Artificial Intelligence*. https://doi.org/10.5220/0005757204380445

Cao, H., Wei, L., Chen, M., Zhou, W., & Hu, S. (2023). Are Large Language Models Good Fact Checkers: A Preliminary Study. *arXiv preprint arXiv:2311.17355*.

Cao, Y., Sun, Y., Xu, C., Li, C., Du, J., & Lin, H. (2022). CAILIE 1.0: A dataset for Challenge of AI in Law - Information Extraction V1.0. *AI Open*, *3*, 208-212. https://doi.org/https://doi.org/10.1016/j.aiopen.2022.12.002

Chen, Z. (2023). PALR: Personalization Aware LLMs for Recommendation. *arXiv preprint arXiv:2305.07622*.

Choi, H., Kim, J., Joe, S., & Gwon, Y. (2021). Evaluation of BERT and ALBERT Sentence Embedding Performance on Downstream NLP Tasks. In *25th International Conference on Pattern Recognition (ICPR)* (pp. 5482-5487). IEEE Computer Society. https://doi.org/10.1109/icpr48806.2021.9412102

Coyne, S., Sakaguchi, K., Galvan-Sosa, D., Zock, M., & Inui, K. (2023). Analyzing the Performance of GPT-3.5 and GPT-4 in Grammatical Error Correction. *ArXiv*, *2303*. https://doi.org/10.48550/arXiv.2303.14342

Criado, J.I., & Gil-Garcia, J.R. (2019). Creating public value through smart technologies and strategies: From digital services to artificial intelligence and beyond, *International Journal of Public Sector Management*, 32(5), 438-450. https://doi.org/10.1108/IJPSM-07-2019-0178

Dale, R. (2020). Natural language generation: The commercial state of the art in 2020. *Natural Language Engineering*, *26*(4), 481-487. https://doi.org/10.1017/S135132492000025X

Davies, M.J. (2020). *The Legal English Grammar Guide*. Self-published.

Doan, A., Ramakrishnan, R., & Vaithyanathan, S. (2006). *Managing information extraction: state of the art and research directions* Proceedings of the 2006 ACM SIGMOD international conference on Management of data, Chicago, IL, USA. https://doi.org/10.1145/1142473.1142595

Dong, C., Li, Y., Gong, H., Chen, M., Li, J., Shen, Y., & Yang, M. (2022). A Survey of Natural Language Generation. *ACM Comput. Surv.*, *55*(8), Article 173. https://doi.org/10.1145/3554727

Espenschied, L.E. (2011). *The Grammar and Writing Handbook for Lawyers*. American Bar Association.

Fedus, W., Dean, J., & Zoph, B. (2022). A Review of Sparse Expert Models in Deep Learning. *ArXiv e-prints*, arXiv: 2209.01667.

Fitsilis, F. (2021). Artificial Intelligence (AI) in parliaments–preliminary analysis of the Eduskunta experiment. *The Journal of Legislative Studies*, 27(4), 621-633. https://doi.org/10.1080/13572334.2021.1976947

Fitsilis, F., & de Almeida, P. (2023) Artificial Intelligence and its Regulation in Representative Institutions. In *Research Handbook on Public Management and Artificial Intelligence*. Cheltenham, UK: Edward Elgar Publishing.

Fitsilis, F., & Makropoulou, G. (2022). *Nomoparaskeuastiki Diadikasia kai Nomiki Pliroforiki*. Nomiki Bibliothiki, Athens.

Fitsilis, F., & Mikros, G. (2022). *Smart Parliaments: Data-driven Democracy*. European Liberal Forum: Brussels, Belgium.

Fitsilis, F., & Papastylianou, A. (2022). Structuring from scratch a legal interoperability laboratory. Revista Política Austral, 1(1), 69-86. https://doi.org/10.26422/RPA.2022.0101.fit

Fitsilis, F., & Papastylianou, A. (2023). Training of Greek Public Administrators in Legal Knowledge Management by Using the Legislation Editing Open Software (LEOS). Journal of the Knowledge Economy, 1-14. https://doi.org/10.1007/s13132-023-01429-z

Fitsilis, F., von Lucke, J., Mikros, G., Ruckert, J., Alberto de Oliveira Lima, J., Hershowitz, A., Philip Todd, B., & Leventis, S. (2023). Guidelines on the Introduction and Use of Artificial Intelligence in the Parliamentary Workspace (Version 1). figshare. https://doi.org/10.6084/m9.figshare.22687414.v1

Flachs, S., Lacroix, O., & Søgaard, A. (2019). Noisy Channel for Low Resource Grammatical Error Correction. In H. Yannakoudakis, E. Kochmar, C. Leacock, N. Madnani, I. Pilán, & T. Zesch (Eds.), *Proceedings of the Fourteenth Workshop on Innovative Use of NLP for Building Educational Applications* (pp. 191-196). Association for Computational Linguistics. https://doi.org/10.18653/v1/W19-4420

Gadyatskaya, O., & Papuc, D. (2023). ChatGPT Knows Your Attacks: Synthesizing Attack Trees Using LLMs. In *International Conference on Data Science and Artificial Intelligence* (pp. 245-260). Singapore: Springer Nature Singapore. https://doi.org/10.1007/978-981-99-7969-1_18

Gallegos, I. O., Rossi, R. A., Barrow, J., Tanjim, M. M., Kim, S., Dernoncourt, F., Yu, T., Zhang, R., & Ahmed, N. K. (2023). Bias and fairness in large language models: A survey. *arXiv preprint arXiv:2309.00770*.

García-Remesal, M., Maojo, V., & Crespo, J. (2010). A knowledge engineering approach to recognizing and extracting sequences of nucleic acids from scientific literature. *Annu Int Conf IEEE Eng Med Biol Soc*, *2010*, 1081-1084. https://doi.org/10.1109/iembs.2010.5627316

Garner, B. A. (2001). *A dictionary of modern legal usage*. Oxford University Press, USA.

Gotti, M. (2014). Linguistic Insights into Legislative Drafting. *The Theory and Practice of Legislation, 2(2)*, 123-143. https://doi.org/10.5235/2050-8840.2.2.123

Han, N.-R., Chodorow, M., & Leacock, C. (2006). Detecting errors in English article usage by non-native speakers. *Natural Language Engineering*, *12*(2), 115-129. https://doi.org/10.1017/S1351324906004190

Harispe, S., Ranwez, S., Janaqi, S., & Montmain, J. (2015). *Semantic Similarity from Natural Language and Ontology Analysis*. Springer Cham. https://doi.org/10.1007/978-3-031-02156-5

Harper, C.M., & Zhang, S.S. (2021). Legal Tech and Lawtech: Towards a Framework for Technological Trends in the Legal Services Industry. In: Gimpel, H., et al. *Market Engineering*. Springer, Cham. https://doi.org/10.1007/978-3-030-66661-3_11

Harris, M., & Wilson, A. (2024). Representative Bodies in the AI Era: Insights and Recommendations for Legislatures (Volume 1). POPVOX Foundation, US.

Hirota, W., Suhara, Y., Golshan, B., & Tan, W.-C. (2020). Emu: Enhancing multilingual sentence embeddings with semantic specialization. In *Proceedings of the AAAI Conference on Artificial Intelligence* (Vol. 34, pp. 7935-7943).

Hohpe, G., & Woolf, B. (2004). *Enterprise integration patterns: Designing, building, and deploying messaging solutions*. Addison-Wesley Professional.

Hong, J., Voss, C., & Manning, C. (2021). Challenges for Information Extraction from Dialogue in Criminal Law. In A. Field, S. Prabhumoye, M. Sap, Z. Jin, J. Zhao, & C. Brockett (Eds.), *Proceedings of the 1st Workshop on NLP for Positive Impact* (pp. 71-81). Association for Computational Linguistics. https://doi.org/10.18653/v1/2021.nlp4posimpact-1.8

Hradec, J., Ostlaender, N., Macmillan, C., Acs, S., Listorti, G., Tomas, R., & Arnes Novau, X. (2019) Semantic Text Analysis tool: SeTA. Publications Office of the European Union, Luxembourg. https://doi.org/10.2760/577814

Huang, J., Gu, S. S., Hou, L., Wu, Y., Wang, X., Yu, H., & Han, J. (2022). Large Language Models Can Self-Improve. *ArXiv e-prints*, arXiv: 2210.11610.

Ji, Y., Tong, C., Liang, J., Yang, X., Zhao, Z., & Wang, X. (2019). A deep learning method for named entity recognition in bidding document. *Journal of Physics: Conference Series*, *1168*(3), 032076. https://doi.org/10.1088/1742-6596/1168/3/032076

Jiang, H., He, P., Chen, W., Liu, X., Gao, J., & Zhao, T. (2020). SMART: Robust and Efficient Fine-Tuning for Pre-trained Natural Language Models through Principled Regularized Optimization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics* (pp. 2177-2190). Association for Computational Linguistics. https://doi.org/10.18653/v1/2020.acl-main.197

Kang, W.-C., Ni, J., Mehta, N., Sathiamoorthy, M., Hong, L., Chi, E., & Cheng, D. Z. (2023). Do LLMs Understand User Preferences? Evaluating LLMs On User Rating Prediction. *arXiv preprint arXiv:2305.06474*.

Khare, A., Dutta, S., Li, Z., Solko-Breslin, A., Alur, R., & Naik, M. (2023). Understanding the Effectiveness of Large Language Models in Detecting Security Vulnerabilities. *arXiv preprint arXiv:2311.16169*.

Kim, J., Derakhshan, A., & Harris, I. G. (2023). Robust Safety Classifier for Large Language Models: Adversarial Prompt Shield. *arXiv preprint arXiv:2311.00172*.

Kojima, T., Gu, S. S., Reid, M., Matsuo, Y., & Iwasawa, Y. (2022). Large language models are zero-shot reasoners. *Advances in neural information processing systems*, *35*, 22199-22213.

Koryzis, D., Dalas, A., Spiliotopoulos, D., & Fitsilis, F. (2021). Parltech: Transformation framework for the digital parliament. *Big Data and Cognitive Computing*, 5(1), 15. https://doi.org/10.3390/bdcc5010015

Lauriola, I., Lavelli, A., & Aiolli, F. (2022). An introduction to Deep Learning in Natural Language Processing: Models, techniques, and tools. *Neurocomputing*, *470*, 443-456. https://doi.org/https://doi.org/10.1016/j.neucom.2021.05.103

Lee, J. (2004). Automatic Article Restoration. In *Proceedings of the Student Research Workshop at HLT-NAACL 2004* (pp. 31-36). Association for Computational Linguistics. https://aclanthology.org/N04-2006

Lee, N., Li, B. Z., Wang, S., Yih, W.-t., Ma, H., & Khabsa, M. (2020). Language Models as Fact Checkers? In C. Christodoulopoulos, J. Thorne, A. Vlachos, O. Cocarascu, & A. Mittal (Eds.), *Proceedings of the Third Workshop on Fact Extraction and VERification (FEVER)* (pp. 36-41). Association for Computational Linguistics. https://doi.org/10.18653/v1/2020.fever-1.5

Levagin, B., Lange, K., Walprecht, S., Gerls, F., & Kuehnhenrich, D. (2022). Vereinfachtes Verfahren zur interaktiven Schätzung des Erfüllungsaufwands mittels maschinellen Lernens. *WISTA-Wirtschaft und Statistik*, *74*(3), 53-67. http://hdl.handle.net/10419/260632

Leventis, S. (2022) Software Tools and Services for the Data-Driven Parliament. In F. Fitsilis & G. Mikros (Eds.), *Smart Parliaments: Data-Driven Democracy* (pp. 19-25). European Liberal Forum, Brussels. https://doi.org/10.53121/ELFTPS4

Leventis, S., Anastasiou, V., & Fitsilis, F. (2020). Application of enterprise integration patterns for the digital transformation of parliamentary control. In *Proceedings of the 13th International Conference on Theory and Practice of Electronic Governance* (pp. 738-741). https://doi.org/10.1145/3428502.3428612

Leventis, S., Fitsilis, F., & Anastasiou, V. (2021). Diversification of Legislation Editing Open Software (LEOS) Using Software Agents—Transforming Parliamentary Control of the Hellenic Parliament into Big Open Legal Data. *Big Data and Cognitive Computing*, 5(3), 45. https://doi.org/10.3390/bdcc5030045

Li, B. (2023). Unlocking Multimedia Capabilities of Gigantic Pretrained Language Models. In *Proceedings of the 1st Workshop on Large Generative Models Meet Multimodal Applications* (pp. 3-4). Ottawa, Canada. https://doi.org/10.1145/3607827.3616846

Li, H., Lan, T., Fu, Z., Cai, D., Liu, L., Collier, N., Watanabe, T., & Su, Y. (2023). Repetition In Repetition Out: Towards Understanding Neural Text Degeneration from the Data Perspective. Thirty-seventh Conference on Neural Information Processing Systems,

Li, J., Sun, A., Han, J., & Li, C. (2022). A Survey on Deep Learning for Named Entity Recognition. *IEEE Transactions on Knowledge and Data Engineering*, *34*(1), 50-70. https://doi.org/10.1109/TKDE.2020.2981314

Liu, B. (2012). Sentiment Analysis: A Fascinating Problem. In *Sentiment Analysis and Opinion Mining* (pp. 1-8). Springer International Publishing. https://doi.org/10.1007/978-3-031-02145-9_1

Liu, M., Lang, B., & Gu, Z. (2017). Calculating Semantic Similarity between Academic Articles using Topic Event and Ontology. *CoRR*, *abs/1711.11508*. http://arxiv.org/abs/1711.11508

Loem, M., Kaneko, M., Takase, S., & Okazaki, N. (2023). Exploring Effectiveness of GPT-3 in Grammatical Error Correction: A Study on Performance and Controllability in Prompt-Based Methods. In (pp. 205-219). Association for Computational Linguistics. https://doi.org/10.18653/v1/2023.bea-1.18

Ma, M., & Wilson, B. (2021). The Legislative Recipe: Syntax for Machine-Readable Legislation. *ArXiv, abs/2108.08678*.

Malmi, E., Krause, S., Rothe, S., Mirylenka, D., & Severyn, A. (2019, November). Encode, Tag, Realize: High-Precision Text Editing. In K. Inui, J. Jiang, V. Ng, & X. Wan, *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)* Hong Kong, China.

Mania, K. (2022) Legal Technology: Assessment of the Legal Tech Industry's Potential. *J Knowl Econ*. https://doi.org/10.1007/s13132-022-00924-z

Maruri, K. (2023). Lawmakers Experiment With ChatGPT to Write Bills. Retrieved from: https://www.governing.com/next/lawmakers-experiment-with-chatgpt-to-write-bills

Meskó, B. (2023). The Impact of Multimodal Large Language Models on Health Care's Future. *J Med Internet Res*, *25*, e52865. https://doi.org/10.2196/52865

Micheler, E., & Whaley, A. (2020). Regulatory technology: Replacing law with computer code. *European Business Organization Law Review*, *21*, 349-377. https://doi.org/10.1007/s40804-019-00151-1

Minghini, M., Kotsev, A., & Granell, C. (2022). A European Approach to the Establishment of Data Spaces. *Data*, *7*(8), 118. https://doi.org/10.3390/data7080118

Mistica, M., Zhang, G. Z., Chia, H., Shrestha, K. M., Gupta, R. K., Khandelwal, S., Paterson, J., Baldwin, T., & Beck, D. (2020). Information Extraction from Legal Documents: A Study in the Context of Common Law Court Judgements. In M. Kim, D. Beck, & M. Mistica (Eds.), *Proceedings of the The 18th Annual Workshop of the Australasian Language Technology Association* (pp. 98-103). Australasian Language Technology Association. https://aclanthology.org/2020.alta-1.12

Mizumoto, T., Komachi, M., Nagata, M., & Matsumoto, Y. (2011, November). Mining Revision Log of Language Learning SNS for Automated Japanese Error Correction of Second Language Learners. In H. Wang & D. Yarowsky, *Proceedings of 5th International Joint Conference on Natural Language Processing* Chiang Mai, Thailand.

Naito, T., Watanabe, R., & Mitsunaga, T. (2023). LLM-based Attack Scenarios Generator with IT Asset Management and Vulnerability Information. In *2023 6th International Conference on Signal Processing and Information Security* (ICSPIS) (pp. 99-103). IEEE. https://doi.org/10.1109/ICSPIS60075.2023.10344019

Nilsson, N. J. (2009). *The quest for artificial intelligence*. Cambridge University Press.

OECD (2023). Regulatory sandboxes in artificial intelligence. OECD Digital Economy Papers, No. 356. OECD Publishing, Paris. https://doi.org/10.1787/8f80a0e6-en

Palmirani, M. (2022). Hybrid AI to Support the Implementation of the European Directive. In *International Conference on Electronic Government and the Information Systems*

Perspective (pp. 110-122). Cham: Springer International Publishing. https://doi.org/10.1007/978-3-031-12673-4_8

Pathak, A. R., Pandey, M., & Rautaray, S. (2021). Topic-level sentiment analysis of social media data using deep learning. *Applied Soft Computing*, *108*, 107440. https://doi.org/10.1016/j.asoc.2021.107440

Pelrine, K., Taufeeque, M., Zając, M., McLean, E., & Gleave, A. (2023). Exploiting Novel GPT-4 APIs. *ArXiv.* https://doi.org/10.48550/arXiv.2312.14302

Perez, E., Kiela, D., & Cho, K. (2021). True few-shot learning with language models. *Advances in neural information processing systems*, *34*, 11054-11070.

Peters, M. E., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., & Zettlemoyer, L. (2018). Deep Contextualized Word Representations. In M. Walker, H. Ji, & A. Stent (Eds.), (pp. 2227-2237). Association for Computational Linguistics. https://doi.org/10.18653/v1/N18-1202

Reimers, N., & Gurevych, I. (2019). Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)* (pp. 3982-3992). Association for Computational Linguistics. https://doi.org/10.18653/v1/D19-1410

Roy, A. (2021). Recent Trends in Named Entity Recognition (NER). *ArXiv*. https://doi.org/10.48550/arXiv.2101.11420

Rozovskaya, A., & Roth, D. (2011, June). Algorithm Selection and Model Adaptation for ESL Correction Tasks. In D. Lin, Y. Matsumoto, & R. Mihalcea, *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies* Portland, Oregon, USA.

Salemi, A., Mysore, S., Bendersky, M., & Zamani, H. (2023). LaMP: When Large Language Models Meet Personalization. *arXiv preprint arXiv:2304.11406*.

Santhanam, S., & Shaikh, S. (2019). A Survey of Natural Language Generation Techniques with a Focus on Dialogue Systems - Past, Present and Future Directions. *CoRR*, *abs/1906.00500*. http://arxiv.org/abs/1906.00500

Senninger, R., & Blom‐ Hansen, J. (2021). Meet the critics: Analyzing the EU Commission's Regulatory Scrutiny Board through quantitative text analysis. *Regulation & Governance*, *15*(4), 1436-1453. https://doi.org/10.1111/rego.12312

Sharma, A., Amrita, Chakraborty, S., & Kumar, S. (2022). Named Entity Recognition in Natural Language Processing: A Systematic Review. In D. Gupta, A. Khanna, V. Kansal, G. Fortino, & A. E. Hassanien, *Proceedings of Second Doctoral Symposium on Computational Intelligence* Singapore.

Sharma, G., & Sharma, D. (2022). Automatic Text Summarization Methods: A Comprehensive Review. *SN Computer Science*, *4*(1), 33. https://doi.org/10.1007/s42979-022-01446-w

Stahlberg, F., & Kumar, S. (2020). Seq2Edits: Sequence Transduction Using Span-level Edit Operations. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)* (pp. 5147-5159). Association for Computational Linguistics. https://doi.org/10.18653/v1/2020.emnlp-main.418

Téllez-Valero, A., Montes-y-Gómez, M., & Villaseñor-Pineda, L. (2005). A Machine Learning Approach to Information Extraction. In A. Gelbukh (Ed.), *Computational Linguistics and Intelligent Text Processing* (pp. 539-547). Springer Berlin Heidelberg.

Vajjala, S., & Balasubramaniam, R. (2022, June). What do we really know about State of the Art NER? In N. Calzolari, F. Béchet, P. Blache, K. Choukri, C. Cieri, T. Declerck, S. Goggi, H. Isahara, B. Maegaard, J. Mariani, H. Mazo, J. Odijk, & S. Piperidis, Proceedings of the Thirteenth Language Resources and Evaluation Conference, Marseille, France.

Van Noordt, C. & Misuraca, G. (2022). Exploratory Insights on Artificial Intelligence for Government in Europe. *Social Science Computer Review*, 40(2), 426–444. https://doi.org/10.1177/0894439320980449

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). Attention is all you need. In *Proceedings of the 31st International Conference on Neural Information Processing Systems* (pp. 6000–6010). Curran Associates Inc.

Vlachidis, A., & Tudhope, D. (2016). A knowledge-based approach to Information Extraction for semantic interoperability in the archaeology domain. *Journal of the Association for Information Science and Technology*, *67*(5), 1138-1152. https://doi.org/https://doi.org/10.1002/asi.23485

von Lucke, J, Fitsilis, F., & Etscheid, J. (2023). Research and Development Agenda for the Use of AI in Parliaments. In *Proceedings of the 24th Annual International Conference on Digital Government Research* (DGO '23). Association for Computing Machinery, New York, NY, USA, 423–433. https://doi.org/10.1145/3598469.3598517

Wang, W., Bi, B., Yan, M., Wu, C., Bao, Z., Peng, L., & Si, L. (2019). StructBERT: Incorporating Language Structures into Pre-training for Deep Language Understanding. *ArXiv*, *abs/1908.04577*.

Wang, X., Jiang, Y., Bach, N., Wang, T., Huang, Z., Huang, F., & Tu, K. (2021). Automated Concatenation of Embeddings for Structured Prediction. In C. Zong, F. Xia, W. Li, & R. Navigli (Eds.), *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)* (pp. 2643-2660). Association for Computational Linguistics. https://doi.org/10.18653/v1/2021.acl-long.206

Wang, Z., Shang, J., Liu, L., Lu, L., Liu, J., & Han, J. (2019). CrossWeigh: Training Named Entity Tagger from Imperfect Annotations. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)* (pp. 5154-5163). Association for Computational Linguistics. https://doi.org/10.18653/v1/D19-1519

Wilson, T., Wiebe, J., & Hoffmann, P. (2009). Recognizing Contextual Polarity: An Exploration of Features for Phrase-Level Sentiment Analysis. *Computational Linguistics*, *35*(3), 399-433. https://doi.org/10.1162/coli.08-012-R1-06-90

Winston, P. H. (1992). *Artificial intelligence* (3rd ed.). Addison-Wesley Publishing Co.

Xiong, D. (2015). Semantic Similarity from Natural Language and Ontology Analysis. *ArXiv*, *abs/1704.05295*.

Xu, J., Liu, X., Yan, J., Cai, D., Li, H., & Li, J. (2022). Learning to break the loop: Analyzing and mitigating repetitions for neural text generation. *Advances in neural information processing systems*, *35*, 3082-3095.

Yadav, V., & Bethard, S. (2018, August). A Survey on Recent Advances in Named Entity Recognition from Deep Learning models. In E. M. Bender, L. Derczynski, & P. Isabelle, *Proceedings of the 27th International Conference on Computational Linguistics* Santa Fe, New Mexico, USA.

Yamada, I., Asai, A., Shindo, H., Takeda, H., & Matsumoto, Y. (2020). LUKE: Deep Contextualized Entity Representations with Entity-aware Self-attention. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)* (pp. 6442-6454). Association for Computational Linguistics. https://doi.org/10.18653/v1/2020.emnlp-main.523

Yang, X., He, X., Zhang, H., Ma, Y., Bian, J., & Wu, Y. (2020). Measurement of Semantic Textual Similarity in Clinical Texts: Comparison of Transformer-Based Models. *JMIR Med Inform*, *8*(11), e19735. https://doi.org/10.2196/19735

Yang, Y., Wu, Z., Yang, Y., Lian, S., Guo, F., & Wang, Z. (2022). A Survey of Information Extraction Based on Deep Learning. *Applied Sciences*, *12*(19), 9691. https://www.mdpi.com/2076-3417/12/19/9691

Yuan, Z., & Briscoe, T. (2016, June). Grammatical error correction using neural machine translation. In K. Knight, A. Nenkova, & O. Rambow, *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies* San Diego, California.

Zadgaonkar, A., & Agrawal, A. J. (2021). An overview of information extraction techniques for legal document analysis and processing. *International Journal of Electrical and Computer Engineering*, *11*, 5450-5457.

Zhao, W. X., Zhou, K., Li, J., Tang, T., Wang, X., Hou, Y., Min, Y., Zhang, B., Zhang, J., & Dong, Z. (2023). A survey of large language models. *arXiv preprint arXiv:2303.18223*.

Zhu, L., Xu, M., Bao, Y., Xu, Y., & Kong, X. (2022). Deep learning for aspect-based sentiment analysis: a review. *PeerJ Comput Sci*, *8*, e1044. https://doi.org/10.7717/peerj-cs.1044

Zhuang, B., Liu, J., Pan, Z., He, H., Weng, Y., & Shen, C. (2023). A survey on efficient training of transformers. *arXiv preprint arXiv:2302.01107*.

Appendices

Appendix I - Original list of smart functionalities

#	Category (alpha)	Smart functionality
1	Verification – Correct usage of …	Citations
2		Existing references
3		Acronyms, organisations and other abbreviations
4	Verification – Context aware correct usage of …	Validity and relevance of references
5		Existing legal definitions
6		Specific legal lexicon
7	Granular change tracking – Comparison of documents	Modifications
8		Change Tracking
9	Linguistics support	Use correct linguistic formulations within the structure of the document
10		Correct formulation in accordance with the English Style Guide
11		Detect divergences between different linguistic translations
12		Suggest linguistic formulations in provisions
13	Legal Assistance – within the act	Detect and avoid structures that could create issues in legal interpretation
14		Correlation between recitals and the enacting terms
15		Between previous acts and the new one
16		Incompatibilities in temporal parameters
17		Explicit or implied obligations
18		Detect implicit or incomplete modifications
19		Detect obligations, rights, permissions, penalties

20	Legal Assistance – within the legal corpus	Automatically identify existing legislation relevant for the act under development
21		Identify hidden semantic correlations
22		Detect suspended, repealed, derogated, delegation of power
23		Passive and active references
24		Life cycle of an article
25	Support 'automatic' legal drafting	Drafting transitional measures
26		Large Language Model (LLM) based legal text generation
27		Construct the consolidation text applying amendments
28	Policy dimension	Measure impact of a legislative act
29		Consistency in definitions
30		Repository of legal knowledge
31		Cluster legislative documents
32	Discovery of Practices/Enabling	Automatically extract metadata
33		Classification of corrigenda
34		Discover concrete practices of different styles of drafting

Appendix II - Tools and companies that specialise in Legal Data using LLMs.

Company / Product	Use Case	Link to Source
	Summarising	
Summize	Using ChatGPT to auto-generate contract summaries	https://bit.ly/3KiLAsL
Docket Alarm	Using GPT-3.5 to auto-summarise PDF litigation filings in dockets	https://bit.ly/3Z612wx
Predictice	This French legal research product uses ChatGPT to generate automatic summaries of court decisions	https://bit.ly/3Y6whXk
	Drafting	
DocDraft	Uses GPT to turn client notes and previous cases into first drafts in minutes	https://www.docdraft.ai/
Spellbook by Rally	Uses GPT-3 to review and suggest language for your documents or contracts in Microsoft Word	https://www.spellbook.legal/
Henchman	Uses GPT-3.5 to enrich drafting options, for example to change single to plural in a clause or add an element to a clause	https://bit.ly/3Zx0r7I
	Contracting	
Ironclad	Worked with OpenAI to create an automated redline feature, with GPT-3 automatically generating clause suggestions and redlines in contracts that users can accept or reject with one click	https://bit.ly/3EiR9mS
Lexion Contract Assist	Uses GPT-3 to help lawyers draft, negotiate, and summarise contract terms. Contract Assist auto-generates clause language, inserts clauses from a playbook, produces suggested redlines, and summarises clause language	https://bit.ly/3XBspgI
Contract Works by Onit	Contract Works has developed two new features, Clause Creator and Simplify, using GPT-3. *Clause Creator* auto-generates a clause during the redlining phase based on user specifications. *Simplify* takes any clause and reproduces it in simpler language to reduce "legalese" and complexity	https://bit.ly/3k4QwXw
AxDraft by Onit	Interestingly, AxDraft, another Onit company, has also created two features using GPT – one called Clause Creator and one called Simplify. They do the same thing as the equivalent features in ContractWorks	https://bit.ly/3KfWMGt
Arteria	Canadian CLM company Arteria is using GPT technology in parts of its end-to-end contracting solution	https://www.arteria.ai/

Malbek	Uses GPT-3 to rewrite clauses in plain English to make contracts more accessible to business users in the contract lifecycle	https://www.malbek.io/
Legal Research		
Jurisage MyJr	MrJr is a JV between Jurisage and AltaML using GPT-3.5 to allow users to ask a legal research question in plain language and get a quick, plain language answer back that synthesises case law	https://bit.ly/3XFZ2tM
Lexata	Uses GPT 3.5 to provide clear, accurate answers to complex securities law questions. Lexata draws on a curated database of securities laws to generate answers and surfaces the relevant sections of securities law to users alongside the answer	https://bit.ly/3IbZWZh
Blue J Legal	Blue J is developing a new research product called “Ask Blue J” that will launch soon and uses ChatGPT across curated, current tax and legal domain content, cross-referencing the answers with legitimate sources to produce a bespoke memo on a user’s specific query	http://www.askbluej.com/
Alexsei	Using GPT-3.5 (in combination with other models) to respond to research queries by aggregating and summarising relevant sources and auto-generating a memo	https://bit.ly/3YYCjtO
ScotusAI by Standd	Uses GPT-3 and a curated database of SCOTUS opinions to provide a chatbot that users can ask any legal question in order to find out what the Supreme Court has said about it	https://standd.io/scotusai
Classification / Tagging		
Fastcase, Docket Alarm	Using LLM to programmatically extract classification tags for use cases and then populating a database so that the tags can be applied for use in downstream tasks	Status: experimenting
SALI Alliance	Using LLM to programmatically extract classification tags for use cases and then populating a database so that the tags can be applied for use in downstream tasks	Status: experimenting
Search / Knowledge Management		
Standd	A new start-up using GPT-3 to search across lawyers’ own internal work product in response to a query, helping lawyers find the most relevant paragraphs in their previous work and providing links to the documents they are part of as well as contextual information	https://standd.io/
Virtual Assistant / Multiple Use Cases		

Casetext CoCounsel	An AI legal assistant based on the most advanced OpenAI LLM to which lawyers can delegate work including legal research, reviewing documents, preparing for a deposition, reviewing and analysing contracts, extracting data from contracts, reviewing documents for compliance with company's policies, searching through databases & retrieving work products	https://bit.ly/3y87B68
Robin AI	RobinAI is a legal industry platform that uses Large Language Models (LLMs), including Anthropic's model "Claude", to automate tasks such as reading, writing, and editing text data. The platform integrates these models to analyse contracts and suggest real-time edits based on clients' preferences, enabling efficient review and approval processes. While acknowledging the potential for AI errors	https://cutt.ly/gwijcRyQ
LawDroid Co-Pilot	Uses GPT-3 to provide a virtual assistant that can research legal issues, help draft emails and letters, summarise documents, translate, or "just have a chat"	https://bit.ly/3YYARrm
Legal NLP	State-of-the-art software + pre-trained legal-specific models	https://www.johnsnowlabs.com/legal-nlp/
Harvey	Harvey, the tool rolled out by Allen & Overy, apparently uses GPT-4 to help lawyers automate contract analysis, due diligence, conduct research, and generate insights, recommendations, and predictions across multiple practice areas	https://bit.ly/3k6w2xv

Appendix III - Raw data on the prioritisation of smart functionalities

Smart functionalities

Interviewees	#1	#2	#3	#4	#5	#6	#7	#8	#9	#10	#11	#12	#13	#14	#15	#16	#17	#18	#19	#20	#21	#22	#23	#24	#25	#26	#27	#28	#29	#30	#31	#32	#33	#34
#1																	o		o	o						o		o						
#2														o	o					o		o				o	o		o			o		
#3									o				o													o	o		o					
#4	o	o	o	o	o	o			o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o										
#5																			o	o										o		o		
#6			o						o	o	o	o		o	o										o		o		o		o			
#7					o	o								o						o														
#8			o			o	o	o		o																								
#9							o	o	o	o	o	o	o	o											o	o								o
#10	o	o	o				o				o	o	o	o	o	o			o	o			o	o								o	o	o
#11																																		

Note: #11 - no response

Appendix IV - Attributes of categories (gamma)

<table>
<tr><th></th><th>UX</th><th>Business value</th><th>Technology stack</th><th>Data sets</th><th>Performance</th></tr>
<tr><td>Legal verification</td><td>Non-intrusive;
Proactive & reactive</td><td>Quality;
Validated outcome</td><td>UX event handlers;
Asynchronous API calling & throttling</td><td>Local & external repos of legal data</td><td>Disruption of drafting during verification</td></tr>
<tr><td>Change tracking</td><td>Side by side comparisons or inline visualisation</td><td>Unnoticed accidental changes</td><td>Aligned front-end technologies;
3rd party visualisations</td><td>Existing current and past versions of drafted legal data</td><td>Support for large documents</td></tr>
<tr><td>Linguistic support</td><td rowspan="6">Non-intrusive;
Proactive & reactive;
Suggestions in modal form / side panels</td><td rowspan="2">Quality;
Validated outcome</td><td rowspan="2">UX event handlers;
Asynchronous API calling & throttling</td><td>Linguistic ref data</td><td rowspan="3">Disruption of drafting during data set retrievals</td></tr>
<tr><td>Legal assistance</td><td>Legal ref data
Cross referencing</td></tr>
<tr><td>Automated drafting</td><td>Consistency;
Out of date data</td><td>On-premise LLM;
Trusted vendor via secure API</td><td>Repositories of prebuilt templates & amendments</td></tr>
<tr><td>Policy dimension</td><td>Different policy perspectives</td><td rowspan="2">Heavy processing in the business layer; API endpoint acting as aggregator to multiple repositories</td><td>Policies & guidelines repos</td><td>Infrastructure workload</td></tr>
<tr><td>Legal practices</td><td>Quality;
Algo reviewing;
Review efforts</td><td>Predefined suggestions & patterns repos</td><td>Disruption of drafting during verification</td></tr>
</table>

Appendix V - Core technologies associated with smart functionalities

SF	Title	Category (gamma)	Technology
#1	Citations	Verification	Named Entity Recognition
#2	Existing references	Verification	Named Entity Recognition
#3	Acronyms, organisations and other abbreviations	Verification	Named Entity Recognition
#4	Validity and relevance of references	Verification	Named Entity Recognition
#5	Existing legal definitions	Verification	Legal Ontology and Terminology Management
#6	Specific legal lexicon	Verification	Legal Ontology and Terminology Management
#7	Modifications	Change tracking	Advanced Language Editing and Correction
#8	Change Tracking	Change tracking	Advanced Language Editing and Correction
#9	Use correct linguistic formulations within the structure of the document	Linguistic support	Advanced Language Editing and Correction
#10	Correct formulation in accordance with the English Style Guide	Linguistic support	Advanced Language Editing and Correction
#11	Detect divergences between different linguistic translations	Linguistic support	Semantic Similarity
#12	Suggest linguistic formulations in provisions	Linguistic support	Advanced Language Editing and Correction
#13	Detect and avoid structures that could create issues in legal interpretation	Legal assistance	Advanced Language Editing and Correction
#14	Correlation between recitals and the enacting terms	Legal assistance	Semantic Similarity
#15	(Correlation) between previous acts and the new one	Legal assistance	Semantic Similarity
#16	Incompatibilities in temporal parameters	Legal assistance	Semantic Similarity

#17	Explicit or implied obligations	Legal assistance	Information Extraction
#18	Detect implicit or incomplete modifications	Legal assistance	Advanced Language Editing and Correction
#19	Detect obligations, rights, permissions, penalties	Legal assistance	Information Extraction
#20	Automatically identify existing legislation relevant for the act under development	Legal assistance	Semantic Similarity
#21	Identify hidden semantic correlations	Legal assistance	Semantic Similarity
#22	Detect suspended, repealed, derogated, delegation of power	Legal assistance	Information Extraction
#23	Passive and active references	Legal assistance	Named Entity Recognition
#24	Life cycle of an article	Legal assistance	Information Extraction
#25	Drafting transitional measures	Automated drafting	Natural Language Generation
#26	Large Language Model (LLM) based legal text generation	Automated drafting	Natural Language Generation
#27	Construct the consolidation text applying amendments	Automated drafting	Natural Language Generation
#28	Measure impact of a legislative act	Policy dimensions	Information Extraction
#29	Consistency in definitions	Policy dimensions	Semantic Similarity
#30	Repository of legal knowledge	Policy dimensions	Legal Ontology and Terminology Management
#31	Cluster legislative documents	Policy dimensions	Text Classification
#32	Automatically extract metadata	Legal practices	Information Extraction
#33	Classification of corrigenda	Legal practices	Text Classification
#34	Discover concrete practices of different styles of drafting	Legal practices	Text Classification

Funded by the
European Union