TOSL: An Ontology to Detect Abusive Services

Elena Molino-Peña[1,2], José María Cruz-Lorite[1], José María García[1,2] and Antonio Ruiz-Cortés[1,2]

[1]Smart Computer Systems Research and Engineering Laboratory (SCORE), Universidad de Sevilla, Seville, Spain
[2]Research Institute of Informatics Engineering (I3US), Universidad de Sevilla, Seville, Spain

Abstract

The Software as a Service (SaaS) model is a distribution and licensing model governed by a Customer Agreement (CA), which defines the responsibility of the involved parties. Abusive services are characterised by the inclusion of unfair terms in their CAs, resulting in a substantial imbalance between the rights and obligations of the contracting parties, to the detriment of consumers. Legal frameworks such as the EU Directive on Unfair Contract Terms (UCTD) seek to protect consumers from the adverse effects of unilaterally drafted contracts imposed by service providers. However, identifying unfair terms remains a challenging task, requiring meticulous legalese analysis. This process is often time-consuming and error-prone, resulting in a lack of awareness among consumers about the potential risks associated with accepting a service provider's terms of service. This paper introduces the Terms of Service Language ontology (TOSL), a specialised profile of the Open Digital Rights Language (ODRL). TOSL standardises and facilitates the representation of terms of service for online providers by modelling obligations, rights, and prohibitions in a structured format. This approach serves as a foundational layer for the automatic analysis of CAs. We evaluated TOSL by modelling real-world CAs and validating that potentially abusive terms can be identified effectively through SPARQL queries over their TOSL representation.

Keywords

Abusive Services, ODRL, Terms of Service, Ontologies

1. Introduction

On 3 October 2024, the European Commission published the final version of the fitness check in EU consumer law[1]. This assessment reviewed key pieces of EU consumer protection legislation, including the Unfair Contract Terms Directive (UCTD), the Unfair Commercial Practices Directive (UPCD), and the Consumer Rights Directive (CRD), to determine their effectiveness in ensuring a high level of protection in the digital environment. The evaluation included a call for evidence with participation from 14 member states and the UK, surveys involving 10,000 customers and 1,000 companies, and 101 interviews with leading academics and key stakeholders [1].

One of the major issues identified was the widespread presence of unfair terms in service agreements, particularly in online transactions. This problem is exacerbated by the characteristics of the digital context, especially the dematerialisation of contracts and the dispersion of terms across multiple web pages, which make it difficult for average consumers to understand the contractual conditions [2]. Although Customer Agreements are legally binding contracts that define the rights and obligations of both parties [3, 4, 5, 6], consumers rarely read them [7, 8]. This is likely due to their excessive length, complex legal jargon [7, 9], and the consumers' weak bargaining position, which effectively forces them to either accept the terms or forgo access to the service [1, 10]. This widespread neglect is summed up in the phrase often referred to as "the biggest lie on the Web": "I have read and accept these terms"[8].

Despite extensive legislation and sustained efforts by regulatory authorities, ensuring compliance with consumer protection laws remains a persistent challenge [11, 12, 13]. Several approaches have been proposed to address this issue, including machine learning models and semi-automatic classification

OPAL'25: ODRL and beyond: practical applications and challenges for policy-base access and usage control, June 01–02, 2025, Portorož, Slovenia

mmolino@us.es (E. Molino-Peña); joscrulor@alum.us.es (J. M. Cruz-Lorite); josemgarcia@us.es (J. M. García); aruiz@us.es (A. Ruiz-Cortés)
0000-0001-7024-5300 (E. Molino-Peña); 0000-0002-0303-2740 (J. M. García); 0000-0001-9827-1834 (A. Ruiz-Cortés)

[1]https://ec.europa.eu/info/law/better-regulation/have-your-say/initiatives/13413-Digital-fairness-fitness-check

techniques designed to detect unfair terms [8, 10, 14, 15]. While these methods support the automation of detection tasks, they often lack a formal structure for representing and reasoning about unfairness, which limits their interpretability and legal rigour. To overcome these limitations, we propose an ontology-based framework to systematically model and identify unfair terms within a knowledge graph (KG). This structured approach not only enables automated detection but also supports deeper reasoning about contractual conditions.

In this work, we introduce the *Terms of Service Language Ontology* (TOSL), a specialised profile of the Open Digital Rights Language (ODRL). TOSL is designed to extend ODRL by incorporating elements of online contracts that go beyond the conditions governing the use of digital assets, such as legal aspects related to jurisdiction and applicable law. To validate its expressiveness, we manually modelled two real-world agreements, one of Elsevier's agreements and another from OpenAI, and defined a set of SPARQL queries to detect unfair terms. Additionally, we have identified two new types of terms that could be deemed unfair according to the UTCD and the EU consumer law fitness check. The contributions of this study are as follows.

- TOSL, a model that operationalises the structural and legal semantics of customer agreements, enabling automated reasoning over their formalised representation.
- A suite of SPARQL queries to detect potentially unfair terms in any agreement represented using TOSL, aligned with EU consumer protection laws and Lippi et al. guidelines [10].
- An extension of the existing typology of unfair terms, through the identication and formalisation of two new categories that are not yet covered in the literature.

The remainder of the article is organised as follows. Sec. 2 describes the structure of Customer Agreements (CAs), highlighting the need to analyse its parts to discover unfair terms, which are defined in Sec. 3. Sec. 4 presents TOSL in detail, and Sec. 5 validates its use in detecting abusive services. Sec. 6 discusses related work and Sec. 7 summarises our conclusions.

2. SaaS Customer Agreement

The Software as a Service (SaaS) model is a subscription-based method of software distribution governed by a Customer Agreement (CA). The Customer Agreement is a legally binding document or set of documents that specify the rights and obligations of the involved parties concerning the acquisition and use of a service [16]. Although there is no standard structure for SaaS agreements, making comparisons between CAs difficult [4], most providers adopt frameworks that include common elements. Components such as *terms of service (ToS)*, *privacy policy*, *acceptable use policy*, *pricing models*, and *SLAs* are widely recognised and included in these agreements [17, 18].

If we focus on **terms of service**, they establish the provisions and conditions under which services are provided, including rights and responsibilities of both parties, usage policies, confidentiality, intellectual property, renewal, changes, indemnities and termination terms, among other legal aspects [18]. This part often contains the most unfair terms, as it details general conditions such as jurisdiction, governing law, or limitation of liabilities. In the majority of CA, acceptable and unacceptable use by the customer is externalised to a separate **Acceptable Use Policy** document.

Privacy Policy typically outline terms related to compliance with applicable law. They often refer to regulations such as GDPR, CCPA, or HIPAA, as well as security standards like SOC 2 and ISO 27001. Their primary purpose is to inform users and third parties about the collection, processing, and protection of personal data [18]. They can often contain potentially unfair terms regarding the removal of customer data during or after the termination of the agreement.

On the other hand, the **pricing** model typically organises features into distinct tiers, utilising plans and add-ons to facilitate this arrangement [19]. The customer, through a subscription, agrees to pay a periodic fee to obtain the ability to access and leverage the functionality provided by the SaaS [20]. A commonly found unfair term in the pricing is the requirement of unreasonable or excessive overcharges in case the customer breaches the agreement. Automated analysis could be used for this purpose [21].

Concerning the **service level agreement (SLA)**, it regulates the provision of a service by defining a set of guarantees [22]. Each of these describes the commitments that the provider will make to achieve specified levels, usually known as Service Level Objectives (SLOs), for one or more Service Level Indicators (SLIs), which can be computational metrics, such as availability, or non-computational aspects, such as time to resolve support/incident tickets. Optionally, the SLA specifies the compensation (i.e. service credits) to which the customer is entitled in case the objectives are not met [23]. The absence of service credits in an SLA may be considered potentially abusive, as it eliminates the provider's accountability for breaches of the agreement.

In practice, the more complex a service is, the larger and more detailed its CA tends to be. For instance, Google Cloud Platform includes a vast number of clauses in its legal framework, incorporating multiple region-specific provisions and supplementary terms (e.g. Acceptable Use, SLAs, etc.) for the individual services. Services handling sensitive information, such as PayPal, or those that operate under strict SLOs often have particularly extensive and detailed agreements, usually to mitigate liability in breaches. The diversity and fragmentation of documents present an additional challenge. Consider GitHub's CA[2] as a case study: GitHub provides its CA as a collection of PDF documents, starting with the "General Terms", which apply to all of their products. Following these, there are three distinct PDFs that outline "Product Specific Terms" for offerings such as Copilot or Enterprise Server. In addition, there are seven supplementary documents covering topics such as the "GitHub Data Protection Agreement" and the "GitHub Online Services SLA". The pricing is defined on an additional, more user-friendly page. Although GitHub is not among the most complex services, its legal framework is divided into, at least 12 documents, reflecting the inherent difficulty and fragmentation of modern service agreements.

Detecting unfair terms in this context is a significant challenge due to the fragmentation and complexity inherent to these agreements. Each document individually may seem acceptable, but together they could be detrimental to the consumer. This requires a detailed analysis of all the documents involved. Automating this analysis becomes essential, particularly to be able to replicate the analysis with updates in agreements. At this stage, having the ability to define custom criteria for abusiveness and to extend these as the law evolves is highly valuable.

3. Consumer Rights and Unfair Terms

The European Union has a long-standing commitment to consumer protection (in business to consumer relationships), demonstrated through a series of impactful directives, regulations and initiatives over the last decades. It all started with the Unfair Contract Terms Directive in 1993, designed to protect consumers from unfair terms in unilaterally-drafted contracts. This initial step was followed by a succession of initiatives, such as Consumer Sales and Guarantees Directive (1999), Unfair Commercial Practices Directive (2005), and the Consumer Protection Cooperation Regulation (2017), among others. This collection of EU directives and regulations highlights the notable emphasis placed on consumer protection by EU policymakers, underscoring the relevance of works like ours on this topic.

EU legislators operate under the premise that "the consumer is in a weak position vis-à-vis the seller or supplier, both in terms of bargaining power and level of knowledge"[3] and, consequently, requires protection from public institutions. However, despite efforts from the EU, academia and independent projects, abusive practices from suppliers continue to be a significant issue in the digital marketplace. Research has shown that online contracts frequently include unfair terms[24], while controversies over consumer data misuse have fuelled media debates, affecting major companies like Meta and TikTok[13].

These issues have been emphasised in the recently published conclusions of the Digital Fairness Fitness Check. In Annex VI, Section VI.1.6, which addresses unfair contract terms, several long-running problems in the online ecosystem are discussed, including the imbalance in bargaining power between parties, consumers' lack of awareness of contract contents, the prevalence of unfair terms, the use of complex and obscure language, and public concerns regarding how these organisations handle

[2] https://github.com/customer-terms

[3] https://eur-lex.europa.eu/legal-content/EN/TXT/PDF/?uri=CELEX:62010CJ0618

client data. All these challenges are far from being resolved, there is room for improvement, and the empowerment of consumers remains a strong commitment from the EU Commission. This is the context in which this work is framed.

3.1. Types of Unfair Terms

The UCTD stipulates in its third article that a contractual term is unfair (and consequently not binding on consumers) if (1) it has not been individually negotiated and (2), contrary to the requirements of good faith, it causes a significant imbalance in the parties' rights and obligations arising under the contract, to the detriment of the consumer. For greater clarity, Annex I of the UCTD presents an "indicative and non-exhaustive" list of terms which may be regarded as unfair [25]. In this section, we will present the eight types of unfair terms identified in the literature and two new types that we identified following a review of the indicative list of terms outlined in the UCTD. For the remainder of this article, we consider only the eight clauses identified and generally accepted in the literature. However, we have considered it relevant to include the two additional clauses, as we anticipate that they are present in digital marketplace agreements, as illustrated by the following examples. Their inclusion is intended to serve as a basis for future work to provide a collection of concrete examples and case law to support their unfairness.

Loos et al. [24] were the first to identify types or classes of unfair terms by analysing European guidelines, regulations, and court rulings applicable to the digital market. They provided examples of unfair clauses extracted from the terms and conditions of service providers like Google or Facebook. Later, Lippi et al. [10] identified three additional categories of unfair terms and curated a dataset[4] that became the foundation for CLAUDETTE, a tool designed to automatically detect unfair terms. Table 1 shows the types of unfair terms identified by both Loos et al. and Lippi et al., along with the two new types of unfair terms that we propose: Excessive Compensation and Vague Commitment.

Table 1
Types of unfair terms commonly found in online Terms of Service.

Type of term	Label	Definition	Proposed by
Arbitration	A	The obligation for the consumer to submit to arbitration before legal proceedings can be initiated.	Lippi et al. [10]
Choice of Law	LAW	The obligation to resolve disputes according to a law different from the consumer's place of residence.	Loos et al. [24]
Content Removal	CR	The right of the provider to unilaterally remove the consumer's content.	Lippi et al. [10]
Contract by Using	USE	The implicit permission given to the provider by the consumer's acceptance of the agreement by the mere fact of using the service.	Lippi et al. [10]
Excessive Compensation	EC	Consumers' obligation to pay a disproportionately high sum in compensation if they fail to fulfil their duties.	Our contribution
Jurisdiction	J	The obligation to resolve disputes outside the consumer's country of residence.	Loos et al. [24]
Limitation of Liability	LTD	The right of the provider to limit its responsibility for damages or losses incurred by the consumer while using the service.	Loos et al. [24]
Unilateral Change	CH	The right of the provider to unilaterally modify the contract or the service.	Loos et al. [24]
Unilateral Termination	TER	The right of the provider to unilaterally terminate the contract or access to the service.	Loos et al. [24]
Vague Commitment	VC	The provider states their intent but does not establish any real commitments or compensation.	Our contribution

[4] http://claudette.eui.eu/corpora/index.html

For excessive compensation, we refer to the UCTD list of indicative terms, which includes terms "(e) requiring any consumer who fails to fulfil their obligation to pay a disproportionately high sum in compensation" as potentially unfair. For example, Sixt, in their Additional Charges Document[5], includes a term that allows them to charge the consumer €605 if they exceed the maximum mileage specified in the rental contract by more than 100 kilometres. Such a charge may seem excessive for this small excess mileage. The identification of these types of unfair term is complicated without human intervention, as their fairness often depends on the context. In other circumstances, €605 might seem insignificant.

In contrast to Excessive Compensation, which disproportionately punishes the customer, we also consider the opposite situation unfair. This occurs when providers state their intentions but make no firm commitments or establish any compensation for non-fulfilment. We refer to such clauses as Vague Commitment. For instance, in Overleaf[6], we find a non-computational SLI that lacks sufficient binding obligations, as it offers the service of Priority Support without specifying a response time or providing compensation for breaches. Conversely, other companies like Salesforce[7] establish a 15-minute response time for incident handling, setting a clearer standard of accountability.

4. The Terms of Service Language (TOSL)

This section presents the Terms of Service Language (TOSL), an ontology designed to provide a structured and semantically rich representation of SaaS ToS. Built as an extension of the Open Digital Rights Language (ODRL) ontology [26], TOSL focusses on capturing the obligations, rights, and prohibitions defined in real-world ToS documents, while introducing the legal concepts needed to model all the stipulations within an agreement. Figure 1 shows how TOSL should be used in a pipeline, ideally automated, to analyse the agreement.

Given an agreement as input, the first stage involves an iterative process to generate a valid TOSL compliant knowledge graph (KG). We envision this task will be supported by NLP and generative AI techniques for extraction and transformation, combined with syntactic and semantic validators to ensure conformance with the ontology and its logical constraints. With this artefact, the system is prepared to execute various operations, such as detecting obligations, rights or unfair terms, using the solver engine, for instance, an SPARQL-based solver. Additionally, this KG could be leveraged to develop an information retrieval system, inspired by GraphRAG, to build a question answering system for the ToS. Last, the results are processed, verified, and returned as output.

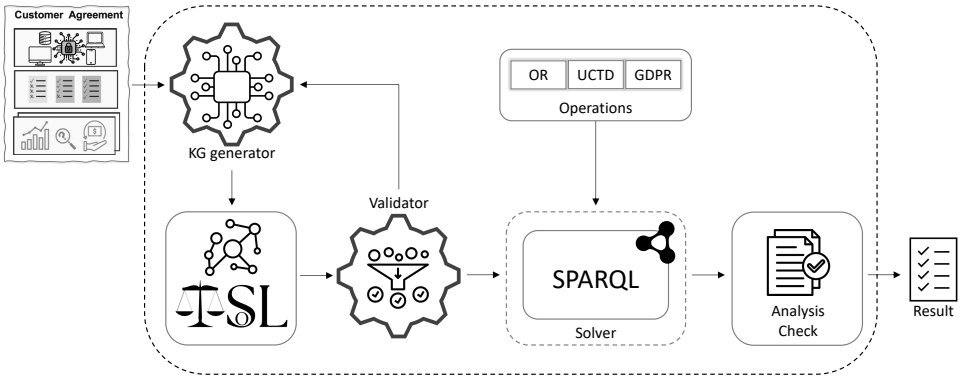

Figure 1: Automatic analysis pipeline for CAs using TOSL to address operations.

In this work, the particular objective is to enable the potentially unfair terms detection operation. The decision to focus on unfair terms in the first version of TOSL is driven by the inherent complexity of

[5]https://www.sixt.es/shared/t-c/EN_Pricelist_additional_charges.pdf
[6]https://www.overleaf.com/user/subscription/plans
[7]https://www.salesforce.com/services/success-plans/overview/signature-success-plans/

CAs, as discussed in Section 2. SaaS providers address many topics in numerous and varied documents, making comprehensive analysis an arduous task. However, unfair terms, particularly in SaaS ToS, are well-studied in existing literature, as outlined in Section 3, with established AI tools available [10, 8, 27, 28, 24], available corpora, and even a partial terminology taxonomy [29]. By building on this foundation, TOSL can leverage existing methodologies to create a robust initial framework, serving as a starting point for future expansions to encompass the broader scope of CA representation and be adaptable to evolving needs. Therefore, the key difference between TOSL and previous approaches is its flexibility to dynamically incorporate new concepts and terminology. This allows for the integration of new types of abuse without the need to manually classify terms or retrain an AI model.

By leveraging the representation of permitted, obligatory and prohibited actions, in addition to detecting unfair terms, it allows for the exploration of other essential questions such as: *What are the obligations of the provider? What are the rights of the consumer? What actions are prohibited?* This contributes to clarifying the responsibilities of the parties. That is, a well-defined representation of the agreement could facilitate the comparison and automation of compliance checks, enabling more sophisticated analysis operations [30, 6]. This structured approach would enhance the ability to assess fairness and legality, supporting better decision-making by the parties involved [4]. The proposed solution is aimed at providers, consumers, government agencies and consumer organisations. Firstly, TOSL enables the formal and unambiguous expression of agreement clauses. Secondly, the formalised clauses can serve as a data model for researchers, practitioners and developers, enabling automated analyses and empowering user-facing applications for non-technical users.

4.1. Methodology

In developing TOSL, we have prioritised principles that balance usability and realism [31]. Usability ensures that the ontology is adaptable and extensible, while realism reflects the accurate representation of real-world ToS terms, including abusive ones. To encourage extensibility, the ontology follows an open design, which allows the representation of any ToS term and supports future extensions of concepts. To achieve these objectives, we based on the NeOn methodology [32]. NeOn emphasises defining functional requirements through Competency Questions (CQs) that align with the ontology's goal, domain vocabulary, and primary use cases, serving as a basis for testing the model.

Both knowledge acquisition and ontology requirements specification activities are central to the design and development of the ontology. Since the general objective of the ontology is to represent contractual terms as deontic modalities, along with sufficient information to identify potentially abusive terms, we identified the UCTD [25] and previous work [10, 24] as non-ontological resources. These resources allowed us to define the initial vocabulary and identify the necessary entities. Subsequently, we adopted ODRL [26], a model and vocabulary designed to represent permitted, mandatory, and prohibited actions. Choosing ODRL as an ontological resource enabled us to build on a validated and interoperable framework. The result of this phase is the TOSL Ontology Requirements Specification Document[8] (ORSD) recommended by NeOn, which includes the purpose, the scope, the implementation language, target group and intended uses of the ontology, and the set of requirements in the form of CQ.

Our CQs are derived from the UCTD's list of terms, the Loos reflection, and the tagging guidelines developed by Lippi et al. for creating the *50 Terms of Service in English Dataset* [10]. These questions are designed to identify potential abuse in the TOS. Consequently, these CQs serve as a benchmark to evaluate whether the ontology provides sufficient information and detail for practical application. Additionally, they ensure that the vocabulary is robust enough to effectively assess the unfairness of the terms. Table 2 presents a selection of CQs defined for each type of unfairness, along with the corresponding derived key concepts. These concepts form the basis for the ontology's conceptualisation. When evaluating a stipulation, an affirmative answer to any of these questions suggests that the term may be potentially unfair. The complete list of CQs is available in the README file of the GitHub repository[9] and in the ORSD document.

[8] https://github.com/isa-group/tosl/blob/main/TOSL_ORSD.pdf
[9] https://github.com/isa-group/tosl

Table 2
Core Competency Questions

Competency Questions	Domain Concepts
A. Is arbitration mandatory before taking any court action?	Arbitration, Litigation, Dispute
LAW. Is the governing law fixed and different from the consumer's country of residence?	Law, Consumer Country
CR. Can the service provider remove the consumer's content?	Remove, Consumer Content
USE. Is the user legally bound by terms just by using the service?	Consent, Implicit, Explicit
J. Does the jurisdiction require dispute resolution outside the consumer's residence?	Jurisdiction, Consumer Courts, Dispute Resolution
LTD. Does the contract attempt to exempt the provider from liability for gross negligence or intentional damage?	Liability, Any Damages, Limit
CH. Is the provider allowed to modify the contract unilaterally?	Change, Prior Notice, justification
TER. Can the provider terminate the contract unilaterally?	Terminate, Justification

Based on the ODRL model and the TOSL ORSD, we performed a reengineering process to adapt and extend ODRL, resulting in the TOSL conceptual model, which is based on the identified terms (see Figure 2) and implemented in Turtle format[10]. Finally, we formalise CQs as SPARQL queries to identify potentially abusive terms included in real-world ToS. The process of formulating CQs based on prior work and translating them into SPARQL queries is thoroughly documented in an annexed explanatory README[11]. In it, for each category of unfairness, we establish the relevant criteria proposed by Lippi et al., identify the corresponding entry in the UCDT, formulate the CQs that enables reasoning about the term's fairness, and finally provide its operationalisation through SPARQL queries.

4.2. The Open Digital Rights Language (ODRL)

ODRL is a policy expression ontology that offers a flexible, interoperable modelling language and common vocabulary to represent statements about the use of digital content and services [26]. The core model of ODRL includes essential concepts such as permissions, prohibitions, and duties, which are vital to articulating the responsibilities typically found in SaaS ToS. In addition, it defines actions, assets, parties, and constraints, facilitating a clear representation of the complex relationships present in these documents. Figure 2 illustrates how these concepts interconnect.

As an overview, in ODRL each policy should consist of a set of rules. Each rule models a statement indicating that a specific action is permitted, mandatory, or prohibited in relation to certain assets governed by the policy. These rules must be executed or adhered to by designated parties. Rules may also include additional elements, such as restrictions (e.g., temporal limits). Furthermore, each type of rule may include an associated duty, which can function in the following manners:

- Precondition. A requirement that must be met before the action is allowed, such as prepayment to access a service.
- Consequence. A reward for not complying with an obligation, such as monetary compensation.
- Remedy. An obligation to take corrective action to compensate for failure to comply with a prohibition.

Crucially, ODRL's extensibility through profiles allows for the introduction of additional semantics tailored to specific needs. Although the core vocabulary provides a strong foundation, ToS often includes complicated terms, such as limitations of liability, jurisdiction, and arbitration, that require specialised concepts. By defining a custom profile, we can extend ODRL to capture these nuances, allowing for more precise modelling of ToS and accommodating terms beyond the core vocabulary.

[10] https://w3id.org/tosl
[11] https://github.com/isa-group/tosl/blob/main/sparql_queries/unfair_terms/README.md

4.3. TOSL Information Model

The main contributions of the Terms of Service Language (TOSL) focus on the representation of responsibilities, their limitations, the differentiation between clauses that apply to B2C and B2B relationships and the modelling of dispute resolution mechanisms along with their corresponding applicable laws and courts. These are concepts and relationships that are difficult to represent using the existing ODRL framework. In addition, TOSL introduces a new domain-specific vocabulary to more accurately represent ToS stipulations. For instance, it includes typical liability exclusions such as *discontinuity*, to indicate whether the provider is liable for service outages, *anyLoss*, to specify responsibility for customer data loss, and *thirdparty*, to clarify liability for damages caused by third parties, among others. It also includes actions, for example, to represent contract termination, content removal, or claims. Moreover, TOSL adds vocabulary to express common restrictions on actions, such as requiring a justification or imposing temporal limits.

Based on the ODRL structure and, guided by the CQs and the most frequent terms, we conceptualised the TOSL model as shown in Figure 2, where the included extensions are represented, in blue the objects and instances, and in bold the relationships. In the conceptual model, we use the prefix *tosl:* to denote the custom extensions introduced by TOSL, while *odrl:* refers to the core elements of the standard ODRL vocabulary. In total, TOSL extension consists of 11 classes, 11 object properties and 41 instances.

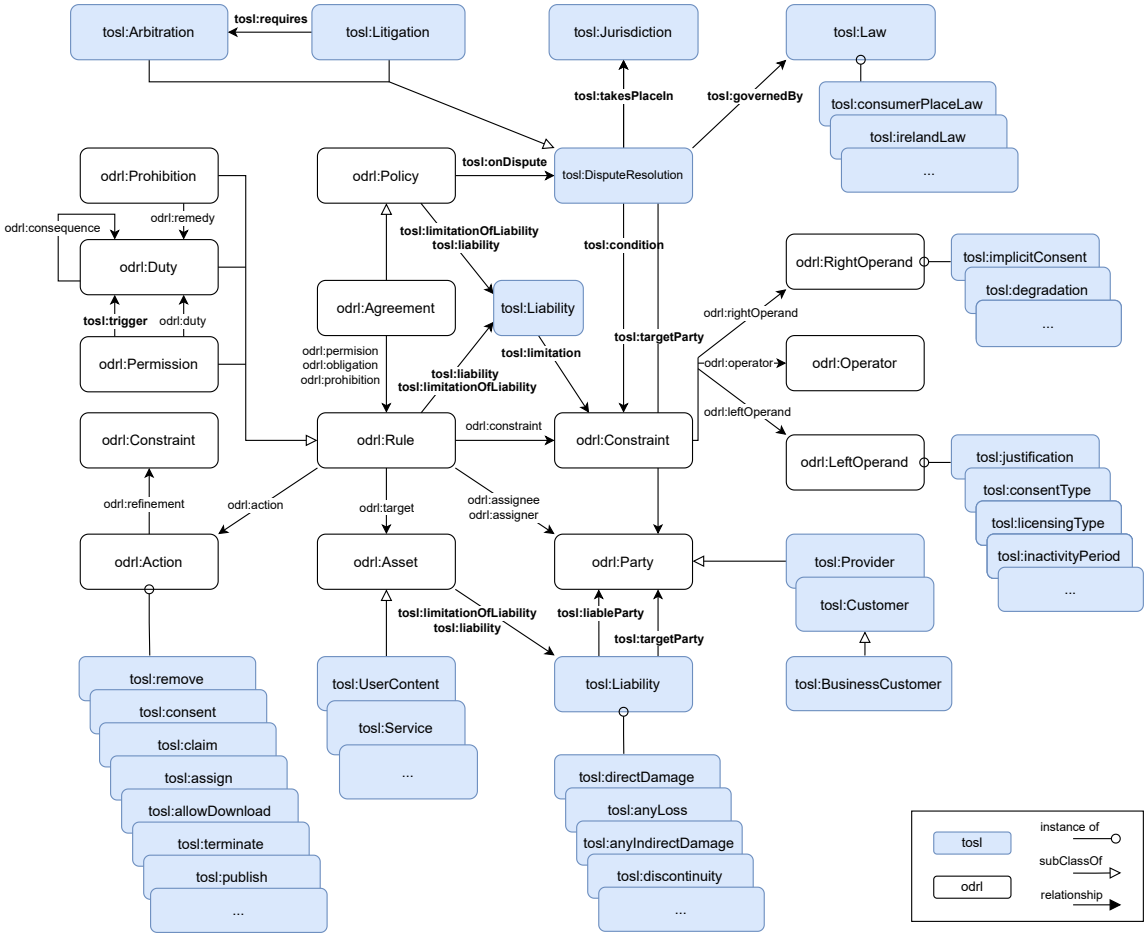

Figure 2: Overview of the TOSL model. The key classes and properties are represented, encompassing domain relationships and ODRL associations.

One of the core features included in TOSL is the party's liability (*tosl:Liability*) that delimits the extent to which a service provider or a customer will be responsible for damages or losses arising from the use of the services. The ODRL framework lacks the capacity to represent the scope of liabilities due

to its design limitations. In this profile, we can represent such terms as either positive obligations or negative limitations. These can directly affect the entire contract, a specific rule, or a particular asset (e.g., a service). Consequently, the domain of the relationships *tosl:liability* and *tosl:limitationOfLiability* can be defined as *tosl:Policy*, *tosl:Rule*, or *tosl:Asset*, while their range is specified as *tosl:Liability*. A liability includes a responsible party (*tosl:liableParte*) and a target party (*tosl:targetParty*) to whom the obligation or exclusion is applicable. This enables filtering provisions that apply specifically to business customers when needed, for example, to exclude those terms from the unfairness assessment, since, as mentioned earlier, legislation only provides protection in B2C relationships.

While liability focusses on the existence of a responsibility, the limitation of liability addresses exclusions, emphasising what the party is not liable for. This distinction is fundamental to understanding contractual terms relating to liability.For instance, a provider may be liable for losing customer data and compensating them monetarily, or simply excluding itself and not being liable. In order to illustrate how a real-world liability would be modeled using TOSL, the following term form GitHub's ToS serves as an example: "GitHub's maximum liability to Customer for any incident giving rise to a claim will not exceed the amount Customer paid for the Product during the 12 months before the incident". In this case, GitHub's liability is limited to a claim amount that does not exceed the total amount paid by the customer during the previous 12 months, as formally represented in the Listing 1.

Listing 1: GitHub Liability and Dispute Resolution Representation

```
# Agreement definition
:agreement01 a odrl:Agreement ;
   odrl:assignee :customer01 ;
   odrl:assigner :github ;
   tosl:liability :liability01 ;
   tosl:onDispute :litigation01 .

# Liability definition
:liability01 a tosl:Liability ;
   dcterms:description "GitHub's maximum liability to Customer for any incident (...) will not exceed the amount
        Customer paid for the Product during the 12 months before the incident" ;
   rdf:value tosl:anyLiability ;
   tosl:liableParty :github ;
   tosl:targetParty :customer01 ;
   tosl:limitation [
         a odrl:Constraint ;
         odrl:leftOperand tosl:totalAmount ;
         odrl:operator odrl:lte ;
         odrl:rightOperand :consumerQuote12Months ;
      ] .

# Dispute resolution definition
:litigation01 a tosl:Litigation;
   dcterms:description "If Customer's principal office is European Union (...) Agreement will be governed by the
        laws of Ireland" ;
   tosl:takesPlaceIn :DublinCourts ;
   tosl:governedBy :IrelandLaw ;
   tosl:targetParty :customer01 ;
   tosl:condition [
         a odrl:Constraint ;
         odrl:leftOperand tosl:consumerResidentCountry ;
         odrl:operator odrl:isAnyOf ;
         odrl:rightOperand ("ES" "DE" "PT" "IT" ...) ;
      ] .

:DublinCourts a tosl:Jurisdiction;
   dcterms:description "Disputes jurisdiction of the Dublin courts" .

:IrelandLaw a tosl:Law ;
   dcterms:description "Governed by the laws of Ireland" .
```

Another main feature of our model is the concept of dispute resolution (*tosl:DisputeResolution*), which is fundamental to modelling the mechanisms through which disagreements or claims between parties are addressed. This component allows the definition of various methods of resolution, including

arbitration (*tosl:Arbitration*) and litigation (*tosl:Litigation*). The litigation process may be preceded by arbitration (*tosl:requires*), providing a layered approach to resolving disputes. The model incorporates provisions on governing law (*tosl:Law*) and includes jurisdiction (*tosl:Jurisdiction*) which are crucial in determining alignment with the legal context of the client's location. Moreover, since dispute resolution methods, applicable law and courts may differ for customers who use the services for commercial purposes, it is necessary to specify the party to which the mechanism applies (*tosl:targetParty*). This comprehensive modelling approach ensures a robust and contextually appropriate representation of dispute resolution processes.

As a real example, GitHub ToS includes a specific dispute resolution term that states: "If Customer's principal office is within the European Union, European Economic Area, or Switzerland, however, this Agreement will be governed by the laws of Ireland. Any legal action or proceeding will be brought exclusively in the courts located in Dublin". This can be represented semantically as follow: the applicable law is defined as the laws of Ireland, the jurisdiction is specified as the Dublin courts and the method of dispute resolution is a formal court procedure. In addition, we can model a condition (*tosl:Condition*) indicating the recipients of this term, specifically Customers whose principal offices are located in the European Union, European Economic Area, or Switzerland, see the Listing 1. By explicitly modelling these aspects, our framework allows us to identify that this term is not potentially abusive in mandatory *Arbitration*, as the specified method of resolution is litigation.

Besides incorporating new classes to enrich the ODRL model representation, the formalisation of the TOSL model requires the addition of new domain-specific vocabulary to complement the existing one. While the ODRL vocabulary includes at least 49 actions for rules, these are not specific enough to define all contractual terms. Therefore, we have introduced a list of actions that are commonly used in SaaS ToS. For example, the action *tosl:remove* is primarily used in contracts to indicate whether the provider has the right to delete content or information from the customer. Another action is *tosl:consent*, typically used to specify that the client must agree to use an asset, which could be conditioned to be implicit or explicit consent through the use of a refinement (odrl:Constraint). Similarly, *tosl:terminate* defines whether the provider or the customer has the authority to terminate the contract or services.

In relation to the rules, assets play a crucial role in identifying potentially abusive terms, as the abusiveness of an action often depends on the object upon which the action is performed. For example, when addressing the CQ: *Is the provider allowed to modify the contract or the services unilaterally?* It becomes essential to identify the objects that the provider is permitted to modify. By defining subclasses of the Asset class (e.g. *tosl:UserContent*), we aim to group assets by type, thereby enabling a more refined categorisation of asset types and facilitating specialised searches. Similarly, specifying the correct party responsible for a rule is essential for certain analysis operations. To support this, we introduced three new types: *Provider*, *Consumer*, and *BusinessCustomer*. This distinction allows us to differentiate between contractual terms in B2C and B2B relationships.

Constraint definitions are completely necessary to restrict the rule. A constraint can be defined as a condition that must be satisfied to enable the execution of a specific action. Certain conditions, often outlined in ToS, are not natively supported by the ODRL specification. In particular, the left operand *tosl:justification* serves as a representative example of such a condition, requiring that specific actions be performed only when accompanied by an appropriate justification such as *tosl:degradation*, which could be a perfect constraint to allow some actions to the provider. Lastly, building on this vocabulary expansion, we propose a novel relationship to extend the representation of interdependencies between rules in ODRL. Within the ODRL framework, a permission rule may incorporate an associated duty as a precondition. Similarly, ToS may define obligations that are activated following the execution of a permitted action. To model this scenario, we introduce the trigger relationship (*tosl:trigger*), which links a permission to its corresponding post-execution obligation.

5. Use Case Validation

To validate the ontology, we modelled the Elsevier API Service Agreement and Open AI Terms of Use Agreement[12] using TOSL and applied SPARQL queries to identify unfair terms. The former was selected for its conciseness and comprehensive coverage of unfair terms, while the latter was chosen for its larger scale, broader typology of provisions, and distinction between B2C and B2B relationships. Two authors independently modelled the agreements following ODRL and TOSL best practices, and then consolidated a unified version. The resulting TTL files, available on GitHub[13], were validated with a syntax validator and an ORDL validator[14] to ensure conformance with the ODRL specification.

SPARQL queries, developed from TOSL CQs, are designed to identify potentially abusive clauses. Each type of unfair term is addressed through unique queries, which can be integrated using `AND` operators to ensure a comprehensive identification of all unfair term types. Both the translation of the CQs into the SPARQL query set and the assessment criteria used to determine whether a provision is potentially unfair or not are explained in detail in an annexed document[15].

The queries identified eight potentially unfair terms in the Elsevier and OpenAI agreements, as shown in Table 3. Arbitration clauses were not detected. This type of unfairness is less common in agreements [10, 29], as providers typically allow litigation processes. Focussing on the Elsevier agreement, no content removal terms were found. However, this does not necessarily imply that the contract is fair, as it lacks any mention of what happens to the user's data after the contract ends. Regarding contract formation through use, there is a clause that requires explicit consent to the contract, avoiding unfairness. Meanwhile, the OpenAI clause concerning litigation states that the applicable law and jurisdiction will align with the customer's place of residence, except for business customers, for whom California law and courts apply. Since our SPARQL query filters for B2B clauses, no unfair terms related to choice of law and jurisdiction were detected. The complete execution output for both agreements can be examined in the corresponding Python notebook[16].

Table 3
Number of clauses detected by type

Agreement	A	LAW	CR	USE	J	LTD	CH	TER
Elsevier API Service Agreement	0	1	0	0	1	2	1	3
OpenAI Terms of Use	0	0	1	1	0	4	1	1

Following this, we demonstrate the application of SPARQL queries specifically for analysing *contract by using*, *unilateral termination* and *limitation of liability* terms.

Contract by Using (USE): A clause obliging the consumer to be bound by contractual terms through the use of a service without explicit consent. In TOSL, as shown in the Listing 2, such clauses involving customer consent obligations can be filtered using the RightOperand *tosl:implicitConsent* of a constraint.

Listing 2: SPARQL query to find Contract by Using unfair terms

```
SELECT ?duty ?action ?assignee ?target
WHERE {
   ?duty a odrl:Duty ;
       odrl:action ?action ;
       odrl:assignee ?assignee ;
       odrl:target ?target ;
       odrl:constraint ?constraint .
   ?assignee a tosl:Customer .
   ?constraint odrl:rightOperand tosl:implicitConsent .
   FILTER (?action  = tosl:consent)
}
```

[12]https://dev.elsevier.com/api_service_agreement.html & https://openai.com/policies/eu-terms-of-use/
[13]https://github.com/isa-group/tosl/tree/main/examples
[14]https://odrlapi.appspot.com/
[15]https://github.com/isa-group/tosl/blob/main/sparql_queries/unfair_terms/README.md
[16]https://github.com/isa-group/tosl/blob/main/bin/elsevier_unfair_analysis.ipynb

Unilateral Termination (TER): A termination term is considered abusive when it allows the provider to end the agreement or services unilaterally, without justification or prior notice. In TOSL, the notion of not requiring justification is represented by the absence of an associated *Constraint*. Similarly, the absence of prior notice is captured as the lack of a precondition requiring the provider to inform before exercising the action. As shown in Listing 3, the query filters provider permissions to terminate, suspend, or disable, with no associated *Constraint* or *Duty* to inform. In general, our observations indicate that the most borderline cases involve terms that allow either party to unilaterally terminate the agreement, yet still follow the same problematic patterns. Although the customer has the same right as the provider to unilaterally terminate the agreement, this is still identified as potentially unfair.

Listing 3: SPARQL query to find termination unfair terms

```
SELECT ?permission ?action ?assignee ?target
WHERE {
   ?permission a odrl:Permission ;
           odrl:action ?action ;
           odrl:assignee ?assignee ;
           odrl:target ?target .
   ?assignee a tosl:Provider .
   FILTER (?action  = tosl:terminate || ?action = tosl:suspend || ?action = tosl:disable)
   OPTIONAL {
      ?permission odrl:constraint ?constraint .
      ?constraint odrl:leftOperand tosl:justification
   }
   OPTIONAL {
      ?permission odrl:duty ?duty .
      ?duty odrl:action odrl:inform .
   }
   FILTER (!BOUND(?constraint ) || !BOUND(?duty ))
}
```

Limitation of Liability (LTD): These clauses are identified by filtering entities of type *tosl:Liability*, specifically in cases where the provider explicitly limits its liability or where the liability is subject to a limitation constraint, applied to the customer in general (non-commercial users included). In the first case, a clause directly restricts liability through the property *tosl:limitationOfLiability*. In the second, the clause references the liability and includes a limitation defined via an ODRL constraint. Listing 4 shows the specific query in use.

Listing 4: SPARQL query to find limitation of liability unfair terms

```
SELECT DISTINCT ?liability ?limitationOn ?type ?liableParty
WHERE {
  ?liability a tosl:Liability ;
    tosl:liableParty ?liableParty ;
    tosl:targetParty ?target Party ;
    rdf:value ?type .
  ?target Party a tosl:Customer .
  ?liableParty a tosl:Provider .
  {
    ?limitationOn tosl:limitationOfLiability ?liability .
  } UNION {
    ?limitationOn tosl:liability ?liability .
    ?liability odrl:limitation ?constraint .
  }
  FILTER (?type  IN (tosl:anyLiability, tosl:physicalInjuries, tosl:harmCausedByMalware, tosl:discontinuity,
  tosl:anyIndirectDamage, tosl:directDamage, tosl:anyLoss, tosl:thirdparty,
  tosl:serviceContent, tosl:breachOfContract, tosl:legalCompliance))
}
```

The formal representation of terms in Elsevier and Open AI agreements using TOSL and SPARQL queries allows for efficient identification of unfair terms. A key advantage is the ability to quickly update querying criteria without extra costs, unlike machine learning models reliant on statistical inference. This capability is critical for adapting to changes in legal frameworks, ensuring the system's relevance over time. In comparison, manual modelling is time-consuming and requires prior ODRL

and TOSL knowledge. Applying NLP techniques to translate ToS into TOSL could be a solution. On the other hand, although this approach emphasises a knowledge-centric perspective, focussing on domain expertise rather than technical implementation, our model and vocabulary may still fall short in capturing the nuances of ToS semantics, which could lead to a loss of information needed to identify new abusive terms.

6. Related Work

Early efforts in legal document analysis include projects like the *ToS;DR* initiative [14], which manual evaluates the ToS of various service providers by rating them according to their degree of abusiveness. This uses the EU-funded Open Terms Archive cloud term database [33]. CLAUDETTE [10] emerged as a pioneering tool to automate the detection of unfair terms in ToS. Using natural language processing (NLP) and machine learning (ML) techniques, achieving an F1-score of 78% using a Support Vector Machine (SVM). CLAUDETTE operates on a dataset of 50 online consumer contracts[17], not only identifying unfair terms but also providing rationales to enhance usability and interpretability [28]. Subsequently, LexGlue [34] expanded on this work by introducing a benchmark for legal NLP tasks. Using datasets such as CLAUDETTE's unfair terms, it achieved significantly improved performance (96% F1-score) with models like Legal-BERT.

Parallel to these efforts, frameworks such as the Open Digital Rights Language (ODRL) [26], the European Legislation Identifier (ELI), the European Case Law Identifier (ECLI), and LegalRuleML [35] can be used to specify legal norms, guidelines, and policies, as well as serve as vocabularies. Furthermore, ODRL-based profiles[18] have been developed to formalise statements in various domains. Currently, there are nine publicly known ODRL extensions tailored to specific fields, including the media industry, market data management, and decentralised data storage [36, 37]. In particular, and more aligned with our proposal, some extensions focus on modelling and assessing compliance with privacy and data protection regulations, such as GDPR compliance [38, 39, 40, 41]. These extensions could potentially be integrated with our profile in the future to enhance its functionality and scope.

7. Conclusions and Future Work

The TOSL framework for controlling unfairness in SaaS ToS provides advantages in terms of extensibility and reusability over AI-based approaches. Its knowledge-centric design allows domain experts to incorporate emerging types of unfair terms, enabling precise and targeted queries without the need for retraining or large-scale annotation. Furthermore, the semantic representation of contractual statements enhances TOSL's applicability to other analysis operations, such as identifying a provider's obligations or comparing agreements.

Our experimental results show that TOSL enables the successful execution of SPARQL queries to detect a broad range of potentially unfair terms. The proposed extensions enhance the expressive power of ODRL by incorporating legal concepts such as *liability limitations* and *dispute resolution mechanisms*. However, the framework faces certain limitations in specific scenarios. First, it may lack the vocabulary or relational expressiveness needed to capture more complex contractual nuances. Second, modelling large-scale ToS remains a time-consuming and technically demanding task.

The main challenges ahead include: (i) developing reasoning mechanisms to resolve contradictions between terms, e.g., simultaneously permitting and prohibiting the same action on a given asset and party; (ii) expanding the ontology to cover other critical contract components, such as SLA compensations or data retention policies; and (iii) validating the approach across a broader corpus of SaaS agreements, towards building a benchmark dataset for unfairness analysis in ToS.

[17] http://claudette.eui.eu/corpora/index.html
[18] https://www.w3.org/community/odrl/wiki/ODRL_Profiles

Acknowledgments

This publication is part of the R&D projects PID2021-126227NB-C21, PID2021-126227NB-C22, and PDC2022-133521-I00 funded by MICIU/AEI/10.13039/501100011033/ERDF/EU and of project TED2021-131023B-C21 funded by MICIU/AEI/10.13039/501100011033/European Union NextGenerationEU/PRTR.

Declaration on Generative AI

During the preparation of this work, the authors used ChatGPT to improve grammar and spelling. The authors reviewed and edited the text and are fully responsible for its final content.

References

[1] European Commission, Digital fairness – fitness check on eu consumer law, 2024. URL: https://ec.europa.eu/info/law/better-regulation/have-your-say/initiatives/13413, [Accessed 25-03-2025].

[2] M. Loos, J. Luzak, Update the Unfair Contract Terms Directive for Digital Services, Technical Report PE 676.006, Policy Department for Citizens' Rights and Constitutional Affairs, Directorate-General for Internal Policies, 2021.

[3] R. Funaki, Y. Nagata, K. Suenaga, S. Mori, A contract corpus for recognizing rights and obligations, in: Proc. Language Resources and Evaluation Conference, 2020, pp. 2045–2053.

[4] E. Molino-Peña, J. M. García, Towards a systematic comparison framework for cloud services customer agreements, in: Service-Oriented Computing – ICSOC 2023 Workshops, 2024.

[5] G. Rejithkumar, P. Anish, S. Ghaisas, Automated identification of deontic modalities in software engineering contracts: A domain adaptation-based generative approach, in: IEEE 31st International Requirements Engineering Conference Workshops, 2023, pp. 72–75.

[6] A. Sancheti, A. Garimella, B. Srinivasan, R. Rudinger, What to read in a contract? party-specific summarization of legal obligations, entitlements, and prohibitions, in: Proc. Empirical Methods in Natural Language Processing, 2023, pp. 14708–14725.

[7] European Commission, Fitness Check of EU Consumer Law on Digital Fairness, Commission Staff Working Document SWD(2024) 230 final, European Commission, Brussels, 2024.

[8] J. A. Obar, A. Oeldorf-Hirsch, The biggest lie on the internet: ignoring the privacy policies and terms of service policies of social networking services, Communication & Society 23 (2020).

[9] A. Sancheti, A. Garimella, B. V. Srinivasan, R. Rudinger, Agent-specific deontic modality detection in legal language, in: Proc. Empirical Methods in Natural Language Processing, 2022.

[10] M. Lippi, P. Pałka, G. Contissa, F. Lagioia, H.-W. Micklitz, G. Sartor, P. Torroni, Claudette: an automated detector of potentially unfair clauses in online terms of service, Artificial Intelligence and Law 27 (2019) 117–139.

[11] O. Svensson, The unfair contract terms directive: Meaning and further development, Nordic Journal of European Law 3 (2020) 24–38.

[12] I. V. Kiryushina, T. A. Philippova, Protection of Consumers from Unfair Agreement Terms, Advances in Natural, Human-Made, and Coupled Human-Natural Systems Research, 2023.

[13] M. Durovic, J. Poon, Consumer vulnerability, digital fairness, and the european rules on unfair contract terms: What can be learnt from the case law against tiktok and meta?, Journal of Consumer Policy 46 (2023) 1–25.

[14] H. Roy, Terms of service; didn't read, 2024. URL: https://tosdr.org/, [Accessed 31-07-2024].

[15] S. Dadas, M. Kozłowski, R. Poświata, M. Perełkiewicz, M. Białas, M. Grębowiec, A support system for the detection of abusive clauses in b2c contracts, Artificial Intelligence and Law (2024) 1–39.

[16] Technical-Committee, "Information technology Cloud computing SLA framework-Part 1: Overview and concepts", Tech. Rep. 19086-1:2016(E), ISO/IEC, Geneva, Switzerland, 2016.

[17] J. M. García, O. Martín-Díaz, P. Fernandez, C. Müller, A. Ruiz-Cortés, A flexible billing life cycle for cloud services using augmented customer agreements, IEEE Access 9 (2021) 44374–44389.

[18] J. D. Michels, C. Millard, F. Turton, An analysis of the standard contracts for 40 cloud computing services, Queen Mary School of Law Legal Studies Research Paper No. 334/2020 (2020).
[19] A. García-Fernández, J. A. Parejo, F. J. Cavero, A. Ruiz-Cortés, Racing the market: An industry support analysis for pricing-driven devops in SaaS, in: International Conference on Service-Oriented Computing (ICSOC), 2025, pp. 260–275.
[20] A. García-Fernández, J. A. Parejo, A. Ruiz-Cortés, Pricing4saas: Towards a pricing model to drive the operation of SaaS, in: Forum, Int. Conf. on Advanced Information Systems Engineering, 2024.
[21] A. García-Fernández, J. A. Parejo, P. Trinidad, A. Ruiz-Cortés, Automated Analysis of Pricings in SaaS-based Information Systems, in: International Conference on Advanced Information Systems Engineering (CAISE), 2025. URL: https://arxiv.org/abs/2503.21444. arXiv:2503.21444, In Press.
[22] J. M. García, P. Fernández, C. Pedrinaci, M. Resinas, A. Ruiz-Cortés, Modeling service level agreements with linked USDL agreement, IEEE Trans. on Services Computing 10 (2017) 52–65.
[23] C. Müller, A. M. Gutierrez, P. Fernandez, O. Martín-Díaz, M. Resinas, A. Ruiz-Cortés, "Automated Validation of Compensable SLAs", IEEE Transactions on Services Computing 14 (2021) 1306–1319.
[24] M. Loos, J. Luzak, Wanted: a bigger stick. on unfair terms in consumer contracts with online service providers, Journal of Consumer Policy 39 (2015) 63–90.
[25] Council directive 93/13/eec of 5 april 1993 on unfair terms in consume contracts, Official Journal of the European Communities (1993).
[26] R. Iannella, M. Steidl, S. Myles, V. Rodríguez-Doncel, Odrl version 2.2 ontology, 2024. URL: https://www.w3.org/ns/odrl/2/, [Accessed 25-03-2025].
[27] H.-W. Micklitz, P. Pałka, Y. Panagis, The empire strikes back: Digital control of unfair terms of online services, Journal of Consumer Policy 40 (2017) 367–388.
[28] F. Ruggeri, F. Lagioia, M. Lippi, P. Torroni, Detecting and explaining unfairness in consumer contracts through memory networks, Artificial Intelligence and Law 30 (2021) 59–92.
[29] F. Lagioia, A. Jabłonowska, R. Liepiņa, K. Drazewski, Ai in search of unfairness in consumer contracts: The terms of service landscape, Journal of Consumer Policy 45 (2022).
[30] D. N. Ganapathy, K. P. Joshi, A semantically rich framework to automate cloud service level agreements, IEEE Transactions on Services Computing 16 (2023) 53–64.
[31] J. ElHassouni, A. E. Qadi, Ontology engineering methodologies: State of the art, in: Proceedings of the 5th International Conference on Big Data and Internet of Things, Cham, 2022, pp. 59–72.
[32] M. C. Suárez-Figueroa, A. Gomez-Perez, M. Fernández-López, The neon methodology framework: Ascenario-based methodology for ontologydevelopment, Applied Ontology 10 (2015) 107–145.
[33] European Commission, Open terms archive, 2024. URL: https://opentermsarchive.org/en/, [Accessed 03-08-2024].
[34] I. Chalkidis, A. Jana, D. Hartung, M. Bommarito, I. Androutsopoulos, D. Katz, N. Aletras, LexGLUE: A benchmark dataset for legal language understanding in English, in: Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics, 2022, pp. 4310–4330.
[35] T. Athan, G. Governatori, M. Palmirani, A. Paschke, A. Wyner, Legalruleml: Design principles and foundations, 2015.
[36] N. Fornara, A. Chiappa, M. Colombetti, Using semantic web technologies and production rules for reasoning on obligations and permissions, in: Agreement Technologies, Cham, 2019, pp. 49–63.
[37] S. Steyskal, S. Kirrane, If you can't enforce it, contract it: Enforceability in policy-driven(linked) data markets, in: 11th International Conference on Semantic Systems, 2015, pp. 63 – 66.
[38] S. Agarwal, S. Steyskal, F. Antunovic, S. Kirrane, Legislative compliance assessment: Framework, model and gdpr instantiation, in: Privacy Technologies and Policy, Cham, 2018, pp. 131–149.
[39] M. De Vos, S. Kirrane, J. Padget, K. Satoh, Odrl policy modelling and compliance checking, in: P. Fodor, M. Montali, D. Calvanese, D. Roman (Eds.), Rules and Reasoning, 2019, pp. 36–51.
[40] B. Esteves, H. Asgarinia, A. C. Penedo, B. Mutiro, D. Lewis, Fostering trust with transparency in the data economy era: an integrated ethical, legal, and knowledge engineering approach, in: Proceedings of the 1st International Workshop on Data Economy, 2022, p. 57–63.
[41] B. Esteves, H. J. Pandit, V. Rodríguez-Doncel, Odrl profile for expressing consent through granular access control policies in solid, in: IEEE European Symposium on Security and Privacy, 2021.