RESEARCH-ARTICLE

Is It Worth Using LLMs for Unfair Clause Detection in Terms of Service?

Published: 16 June 2025

ICAIL 2025: 20th International Conference on Artificial Intelligence and Law
June 16 - 20, 2025
IL, Chicago, USA

ICAIL '25: Proceedings of the Twentieth International Conference on Artificial Intelligence and Law (June 2025)
https://doi.org/10.1145/3769126.3769218
ISBN: 9798400719394

Is It Worth Using LLMs for Unfair Clause Detection in Terms of Service?

Marco Panarelli
University of Bologna
Bologna, Italy
marco.panarelli@studio.unibo.it

Andrea Galassi*
DISI
University of Bologna
Bologna, Italy
a.galassi@unibo.it

Francesca Lagioia*
AlmaAI
University of Bologna
Bologna, Italy
European University Institute
Fiesole, Italy
francesca.lagioia@unibo.it

Rūta Liepiņa
AlmaAI
University of Bologna
Bologna, Italy
ruta.liepina@unibo.it

Marco Lippi
DINFO
University of Florence
Florence, Italy
marco.lippi@unifi.it

Przemysław Pałka
Jagiellonian University
Kraków, Poland
przemyslaw1.palka@uj.edu.pl

Giovanni Sartor
AlmaAI
University of Bologna
Bologna, Italy
European University Institute
Fiesole, Italy
giovanni.sartor@unibo.it

Abstract

Unfair clause detection is an extremely useful AI application for consumer protection. Artificial intelligence has recently been successful in building systems capable to automatically detect unfair clauses in Terms of Service, and also to identify their unfairness categories. Since Large Language Models (LLMs) are nowadays bringing a revolution to the field of artificial intelligence, and in particular to natural language processing and understanding, in this paper we compare several different prompt strategies for LLMs with more traditional BERT-based fine-tuned models. Our extensive experimental evaluation aims to investigate whether it is worth using LLMs also for this challenging domain-specific task.

CCS Concepts

• **Computing methodologies** → **Machine learning**; **Natural language processing**; • **Applied computing** → **Law**.

Keywords

Terms of Service, Large Language Models, Transformer-based Models, Unfair Clauses

*Corresponding authors.

ICAIL 2025, Chicago, IL, USA

ACM ISBN 979-8-4007-1939-4/25/06
https://doi.org/10.1145/3769126.3769218

ACM Reference Format:
Marco Panarelli, Andrea Galassi, Francesca Lagioia, Rūta Liepiņa, Marco Lippi, Przemysław Pałka, and Giovanni Sartor. 2025. Is It Worth Using LLMs for Unfair Clause Detection in Terms of Service?. In *20th International Conference on Artificial Intelligence and Law (ICAIL 2025), June 16–20, 2025, Chicago, IL, USA.* ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/3769126.3769218

1 Introduction

Technological advances have enabled many social and economic activities, from communication and shopping to dating and entertainment, to be facilitated by third-party online platforms. Although this shift has been celebrated for reducing transaction costs [2], it has also raised concerns about power imbalances and the erosion of consumer rights in online markets [17, 21, 37]. Accessing these platforms requires users to agree to Terms of Service (ToS) which, like other boilerplate contracts, are drafted unilaterally by service providers. Such ToS often include clauses that are unfavorable to consumers, such as limitations of liability, restrictions on rights in dispute resolution, unilateral change and termination, and concessions of privacy [27, 30, 40].

Even though certain jurisdictions, like the European Union, have enacted laws aimed at eliminating unfair terms from consumer contracts [30, 36], empirical research demonstrates that they have failed to effectively counter such practices so far [27, 29, 41]. Virtually, all ToS of popular online platforms feature clauses directly violating the EU regulations [41]. Such unfair contractual terms, often hidden in lengthy and opaque documents, can impose significant disadvantages on consumers, undermining their rights and their freedom of choice. Against this backdrop, a compelling question

arises: can AI be used to rebalance the scales, empower consumers, and support the enforcement of EU law? By harnessing AI in the service of consumers, there is an opportunity to shift its role from a predominantly corporate asset to a force for accountability and fairness, as a counterpower tool for the society at large [28].

As detailed in Section 2, to explore this potential, a recent trend has emerged in the legal domain, i.e., the use of machine learning methods for the identification and analysis of unfair and unlawful clauses in legally relevant documents. Recent popular approaches are based on supervised learning and widely adopt solutions such as support vector machines, neural networks, and transformer-based models [8]. The problem with this kind of approaches, however, is that their deployment necessitates the laborious creation of datasets for training and evaluation that, over time, might become obsolete – as the language deployed by service providers changes. On the other hand, large language models (LLMs) have recently become a mainstream solution for natural language processing and understanding, making artificial intelligence systems accessible to the large public, and moving huge steps forward, in terms of accuracy and performance. Yet, these models often require very large computational resources to run and, in some cases, the best-performing solutions are proprietary and available on payment only (e.g., ChatGPT or Gemini).

Within this context, the aim of this work is to study whether and to what extent LLMs are capable of performing the task of detecting unfair clauses in ToS. It is, in fact, unclear, whether it is really worth using such models with respect to other transformer-based architectures (such as BERT), which still represent the state-of-the-art for many benchmarks. In principle, LLMs could be used without the necessity to create large training data sets, exploiting their "in-context learning" capability to successfully address a task, even without being explicitly trained for it [6]. Yet, in specific areas, the accuracy of this general-purpose approach could not be sufficient, and other more traditional methods could perform better. We aim to investigate whether the task of unfairness detection in ToS is one of these areas. For this purpose, our contribution is a wide experimental evaluation, comparing 8 classifiers against 7 LLMs, in 6 different settings, for a total of 44 combinations of models and settings. Our experiments focus, in particular, on smaller, open-access, free-to-use models, – Phi3-14B, Nemo-12B, Llama3-8B, LawChat-7B, Mistral-7B – which we compare with transformer models (Legalbert and Deberta-based) as well as with larger LLMs (Codestral-22B, Qwen-32B), given the emerging capabilities of smaller LLMs, especially with few shot-learning [47]. Importantly, the training and use of small models require much fewer resources [33] both environmentally and economically, than larger LLMs, and thus would be more easily accessible by the civil society and by enforcement agencies.

The paper is structured as follows. By way of introduction in Section 2 we set the EU legal framework on unfair contract terms and discuss related works, focusing on the state of the art of natural language processing (NLP) techniques used to address the tasks at hand. In Section 3 we describe the corpus used and the annotation guidelines. Section 4 explains the methodology employed whereas section 5 presents and discusses the experimental results. In Section 6 we conclude with a look to future research.

2 Background

2.1 The EU law on unfair contract terms

Legally speaking, Terms of Service (ToS) of online platforms and mobile apps are contracts [30]. If concluded by consumers, i.e., natural persons acting outside of their profession, ToS are subject to fairness control under the European Union law [36]. Unlike American law, under which provisions potentially adverse to consumers, like mandatory arbitration or distant forum, are generally justiciable [40], European law holds that boilerplate clauses creating a power imbalance to the detriment of consumers are not binding [36]. Examples, under the statutory rules and the jurisprudence of the EU Court of Justice, include limitations of liability for gross negligence, rights to unilaterally modify the contract or the service [29, 30, 36], and other clauses, as detailed in Section 3.

Why would the ability to detect unfair clauses in ToS be socially valuable? First, empirical research shows that consumers do not read the ToS [3, 34, 39] and, thereby, may subscribe to terms they neither know nor expect. A detection system paired with a well-designed user experience could inform the consumers what they agree to when clicking (or tapping) "I accept." As such, this could be a tool for consumer empowerment [28], limiting information asymmetries, and facilitating choice and competition, thereby turning some assumptions of consumer law into reality [4].

Second, unfair clauses can be policed through the so-called abstract control, i.e., the consumer agencies instructing corporations to remove them via administrative procedures [36]. This avenue has become even more promising in the recent years, with the European Union amending its consumer law enforcement procedures as to give the agencies the competence to impose monetary fines on the violators [43]. However, abstract control is laborious and requires a lot of work from the continuously understaffed administrative bodies. Hence, the ability to automate at least part of the unfairness assessment could help consumers, while preserving the thin resources of consumer agencies. For these reasons, research into the automation of unfair clause detection has been booming, as highlighted in the following section.

2.2 Related works

Terms of service (ToS) and privacy policies are essential tools for regulating the contractual relationship between companies and consumers and the processing and protection of their personal data. These documents are among the most popular legal documents studied in the AI and law domain [8]. The reasons for this focus include:

(a) Accessibility: organizations are required to display these documents on their websites, making them easily accessible. Additionally, such documents are often available in multiple languages to facilitate market monitoring, since consumers, protection authorities and non-governmental organisations in Europe tend to operate in their respective languages [15].
(b) Practical impact: automating the analysis of such documents has significant benefits. Manually checking and reviewing platform terms of services and privacy policies is a labor–intensive

and time-consuming process, especially given the frequent updates to these documents and the limited resources available for compliance checking and enforcement.

(c) Consumer empowerment: methods and tools enabling citizens to better understand their rights and the processing of their data can empower them and promote transparency [1, 20, 29, 54].

(d) Support for law enforcement: automating the analysis of terms of services and privacy policies can assist consumer organizations and data protection authorities in their efforts to monitor market practices and enforce compliance with legal requirements.

From the technical perspective, detecting unfairness in these documents has been framed as a detection and classification problem. Analyses of privacy policies [10] primarily focus on identifying common data processing practices, such as data collection, transfer, and sharing. In contrast, analyses of ToS examine key elements of consumer contracts, including termination clauses, changes to terms, and applicable law. Both areas use well-established datasets and have achieved significant advancements through the application of machine learning methods. On the existing public corpora for ToS, the state-of-the-art approaches achieve over 80% precision and recall for the binary task of detecting unfair clauses, using an ensemble of traditional machine learning systems [29], and over 87% using Legal-BERT [18].

The recent experiments with LLMs have shown their capabilities of handling detection and classification tasks, including analysis of legal documents [42, 49, 56], and new benchmarks for legal reasoning with LLMs [19] have been defined, including a dedicated task for the analysis of terms of service. Tang, et al. [50] were one of the first studies to compare LLM abilities to extract information on common data processing practices from privacy policies. The results reported for their PolicyGPT demonstrate an improvement in retrieval metrics, compared to earlier models that relied on expert-annotated datasets. Rodriguez, et al. [44] carried out a comprehensive prompting comparison identifying data types collected by companies. Their study demonstrates that in domains with well-established ground truth (e.g., annotated privacy policy datasets), LLMs can offer efficient and competitive alternatives to traditional methods, while being more cost-effective and time-saving. Optimized prompts achieved performance comparable to state-of-the-art techniques in analyzing privacy policies, using GPT-3 and GPT-4 models, whereas LLama-2 showed weaker results. Tsai et al. [55] have recently used GPT-4o to develop a tool to detect unfavorable financial terms.

As for the terms of services, the classification of unfair terms [29] is part of the LexGLUE benchmark, which has been recently used to test GPT capabilities on legal data [7]. The comparative F1 scores of the zero-shot (micro 41.4, macro 22.2) and few-shot (micro 64.7, macro 32.5) settings with ChatGPT (`gpt-3.5 turbo`) were significantly higher than the random guess baseline (micro 02.9, macro 02.9). However, the general purpose models still fall short of the supervised methods (micro av. 96.0, macro av. 83.0). Concerning languages different from English, the LEXTREME [38] benchmark includes a multilingual version of the task [13], while Bernhard et al. [5] have developed an automatic multilingual scraper for Privacy Policies and Terms of Service.

3 Data

For the purpose of this work, we rely on the CLAUDETTE preexisting dataset, consisting of 142 ToS in English, gradually downloaded from providers' websites since 2017 [24, 29, 45]. These documents were analysed by legal experts and marked in XML by independent annotators. The ToS were selected among those offered by some of the major players in different market sectors, i.e., (i) gaming and entertainment; (ii) social networks and online dating; (iii) travel, accommodation and service intermediaries; (iv) content-sharing platforms; (v) productivity tools and business management; (vi) e-commerce; (vii) search engine and analytics; (viii) health and well-being, (ix) communication tools; (x) finance and payments. [24, 27]. The annotations reflect the methodology described in [24, 29], where nine categories of unfair clauses were identified, i.e., clauses concerning (1) jurisdiction (<j>); (2) applicable law (<law>); (3) liability limitations (<ltd>); (4) the unilateral termination of the contract/service (<ter>); (5) provider's right to unilaterally modify the contract/the service (<ch>); (6) arbitration (<a>); (7) provider's right to unilaterally remove consumer's content (<cr>); (8) consent to the agreement simply by using the service (<use>); and (9) the scope of consent granted to the ToS incorporating also the privacy policy, which forms part of the "General Agreement"(<pinc>). To capture the different degrees of (un)fairness a numeric attribute was included in each label, with 1 meaning *clearly fair*, 2 *potentially unfair*, and 3 *clearly unfair* [24, 29]. Nested tags were used to annotate text segments relevant to more than one type of clause. If one clause covers more than one sentence, each sentence is labeled separately.

In the following, we provide a detailed definition of the mentioned categories of clause and of the conditions under which they can be deemed unfair.

Jurisdiction $\langle j \rangle$: The jurisdiction clause specifies what courts have the competence to adjudicate disputes. A clause is (potentially) unfair whenever it states that judicial proceeding takes a residence away (i.e., in a different city, different country from the consumer place of residence).

Choice of Law $\langle \textbf{law} \rangle$: The choice of law clause specifies what law will govern the contract and be applied in potential disputes. A clause is (potentially) unfair whenever it states that the applicable law is different from the law of the consumer's place of residence the clause is unfair.

Limitation of Liability $\langle \textbf{ltd} \rangle$: The limitation of liability clause specifies for what actions/events and under what circumstances the providers exclude, limit or reduce their liability, the duty to compensate damages and/or when contains a blanket phrase like "to the fullest extent permissible by law". Such clause is always (potentially) unfair, unless it is a force majeure case.

Unilateral Change $\langle \textbf{ch} \rangle$: The unilateral change clause specifies if and under what conditions the provider can unilaterally change and modify the contract and/or the service. Such clause is always (potentially) unfair.

Unilateral Termination $\langle \textbf{ter} \rangle$: The unilateral termination clause states that the provider has the right to suspend and/or terminate the service and/or the contract and/or the consumer's account, due to some reasons, or at any time, for any or no reasons with or without notice. Such clause is always (potentially) unfair.

Table 1: Composition of the final corpus

Type of clause	Tag	#Clauses	#Unfair	#ToS
Arbitration	<a>	165	156	53
Unilateral Change	<ch>	506	506	138
Content Removal	<cr>	261	261	102
Jurisdiction	<j>	218	180	110
Choice of Law	<law>	225	192	130
Limitation of Liability	<ltd>	1072	971	139
Unilateral Termination	<ter>	624	624	134
Consent by Using	<use>	370	370	129
Privacy Included	<pinc>	115	115	80

Contract by Using ⟨use⟩: The contract by using clause states that the consumer is bound by the terms of use/service simply by using the service, downloading the app or visiting the website. Such clause is always (potentially) unfair.
Content Removal ⟨cr⟩: The content removal clause gives the provider a right to modify, delete or remove the user's content, including in-app purchases, under specific conditions or at any time, in his full discretion, for any or no reasons, with or without notice or the possibility to retrieve the content. Such clause is always (potentially) unfair.
Arbitration ⟨a⟩: The arbitration clause requires or allows the parties to resolve their disputes through the arbitration, before the case could go to court. A clause is (potentially) unfair whenever the arbitration is binding and not optional and/or should take place in a country different from the consumer's place of residence and/or be based not on law but on other arbitration rules and/or arbiter's discretion.
Privacy Included ⟨pinc⟩: Identify clauses (a) explicitly stating that, simply by using the service, the consumer consents to the processing of personal data as described in privacy policy; and/or (b) that the privacy policy is incorporated into and form part of the terms and it is preceded by a content by using clause to such terms. These clauses are always (potentially) unfair.

Table 1 reports the composition of the final corpus. It contains 37,895 sentences, where 3,556 are relevant for one or more of the mentioned clause categories. Interestingly, 3,049 out of 3,556 were labeled as containing a potentially or clearly unfair clause. For our work, potentially and clearly unfair clauses were jointly considered as *unfair instances*, whereas the set of *fair instances* contains both clauses that are clearly fair for each relevant category and clauses unrelated to unfairness. The distribution of the different categories across the 142 documents is imbalanced. We observed a high frequency of some of the selected categories within the dataset. Arbitration and privacy included clauses are the most uncommon, being respectively contained only in 53 and 80 documents. All the other categories appear in at least 102 out of 142 ToS. Limitation of liability and unilateral termination together account for approximately half of all the potentially unfair clauses.

4 Method

Building on previous work [29], unfair clause detection can be formally described as a sentence classification problem, within the supervised learning framework. Given a dataset $\mathcal{D} = \{(x_i, y_i\}_{i=1}^{n}$, which consists of n sentences x_i annotated with labels y_i, the goal is to learn a function f that maps x_i into y_i, to be used to predict unfairness of a novel (never seen during training) sentence $\hat{x}$. Function f can be learned, in principle, with any supervised machine learning classifier, whereas sentences x_i can be modeled with different representations, such as bag-of-words or embeddings [26]. As for the labels y_i, unfair clause detection can be considered a multi-label classification problem, since every clause can belong to more than one unfairness category (see Section 3). For example, one clause could be potentially unfair for the consumer both for unilateral change and for content removal categories. Thus, each label y_i is actually a vector of binary variables $y_i = \{y_i^j\}_{j=1}^{m}$, where each element y_i^j is associated to the unfairness for one specific category j out of m (e.g., limitation of liability). If the clause is not unfair for any category, then it is considered fair.

While existing approaches for unfair clause detection have used classic machine learning classifiers, such as support vector machines or neural networks [29], recently transformer architectures like BERT or Legal-BERT have been shown to achieve state-of-the-art performance [8]. In the following, we describe how to address the task with generative AI models.

4.1 Generative AI approaches

There are several ways in which the unfair clause detection task can be formulated for LLMs. Since it is well-known that the performance of LLMs heavily depends on the adopted prompt, we experimented with different prompts. In all cases, we define prompts that are mostly based on the annotation guidelines, i.e., on the definitions of the nine categories of clauses and the conditions of their unfairness, as detailed in Section 3. Indeed, the category definitions provide fundamental information, since the categorys' names may not be informative enough. We implement three strategies for defining prompts, which differ in how they address the tasks of detecting (i.e., recognizing whether a clause is potentially unfair) and classifying it (i.e., identifying the category of unfairness). We name the three strategies *single-prompt*, *multi-prompt*, and *pipeline*. When the proposed strategy allows for it, we experimented both with zero-shot and few-shot learning settings. For the format of the prompt, we take inspiration from previous work [7, 42, 44, 50].

4.1.1 Single-prompt approach. In this approach, both detection and classification are performed simultaneously over a clause, with a single prompt. The prompt uses a comprehensive template that includes all necessary information: the tasks description, the definition of the nine categories as given in Section 3, and detailed instructions for output formatting. The expected output from this prompt is a list containing all the unfairness categories for each input clause. While adequate to zero-shot scenarios, this approach would become cumbersome in a few-shot setting, as it would require incorporating several examples for each category into a single prompt. Such a large context can be difficult to handle for many existing LLMs.

4.1.2 Multi-prompt approach. To address the scalability issues of the single-prompt approach, we decompose the problem into several subtasks. Differently from the previous method each category

is addressed separately. Given an input clause, the model is queried with 9 separate prompts, one for each category, so that it focuses exclusively on one category at a time. This design minimizes the risk of misinterpreting the task requirements or the formatting instructions. In this way, the multi-label classification task is effectively transformed into a set of nine binary classification tasks. This approach supports both the zero-shot and few-shot learning settings, and the prompt template is the following:[1]

```
Evaluate the following clause from a terms of service based on
the provided definition of unfairness:
Definition: {category definition}
If the clause is unfair according to the definition respond with
'yes' otherwise respond with 'no'. Provide only 'yes' or 'no' as
the output. Do not include any explanations or additional text.
Examples: {examples}[2]
Test Input:
Clause: {clause}
Response:
```

4.1.3 Pipeline approach. The third methodology builds on the multi-prompt approach and further decomposes the task into two distinct phases of a pipeline. During the first phase, i.e.,*category identification*, the model determines whether a clause belongs to a specific category. To this aim, we perform one query for each category, using the same definitions of the previous approaches, but omitting the part describing the unfairness conditions. Similarly to the *single-prompt approach*, this phase is always performed in a zero-shot setting for scalability reasons. If all prompts respond negatively (i.e., the clause belongs to none of the categories) the clause is deemed *fair*. The prompt template for this first stage is the following:

```
You are analyzing terms of service. Please, tell whether the clause
talks about the subject specified in the definition.
Definition: {category definition}
Please answer only with 'yes' or 'no'. Do not include any explanations
or additional text.
Test Input:
Clause: {clause}
Response:
```

For clauses associated with one or more categories, we proceed with the second phase, i.e. the *unfairness assessment*. Specifically, if in the first phase a clause is positively identified as belonging to a category, in the second phase, a second query assesses its unfairness for that category. The prompt includes the "unfairness condition" as it appears in the definition of the category, with the exception of those definitions that only state "*Such clause is always unfair*". This may minimize potential biases in the model. The prompt for the second phase is always performed in a few-shot setting. The template is the following:

```
You are analyzing terms of service. Please, tell whether the
input clause is unfair. {unfairness conditions}. If the clause
is unfair respond with 'yes' otherwise respond with 'no'. Provide
only 'yes' or 'no' as the output. Do not include any explanations
or additional text.
Examples: {examples}
Test Input:
Clause: {clause}
Response:
```

Both phases use one prompt per category, for a total of 9 to 18 prompts per sentence.

4.2 Selection of examples for few-shot setting

The multi-prompt and the pipeline approaches are designed to support the few-shot learning settings. In these settings, for each category, the prompt includes 8 examples of unfair clauses. These examples, extracted from the training set, are hand–selected according to the following criteria: (i) coverage of different market sectors; (ii) coverage of diverse types of unfair contracting practices, within each unfair-clause category; (iii) length of clauses.

As a first approach, we select long sentences, since they are usually relevant for a diverse set of unfair practices. For instance, a clause may limit or exclude the provider's liability by (i) liability theory (e.g., tort law, contract law, strict liability, statutory liability, product liability); (ii) causal link with the damage (e.g., special, incidental, direct, indirect, punitive damages); (iii) kind of damage (e.g., economical, reputational); (iv) standard of care (e.g., negligence, gross negligence, awareness); (v) cause of damage (e.g., security breach, computer harms, third parties actions); and (vi) compensation amount (e.g., max 10 euro). Long sentences usually offer several advantages. As noted, they encapsulate a wide range and different combinations of unfair practices, thus providing a richer and more detailed context. This may help the models to capture complex relationships and generalize across diverse scenarios. However, they also come with disadvantages. Small-scale language models may struggle with processing lengthy sentences, due to input length constraints or difficulty in capturing long-range dependencies. Furthermore, the dense nature of long sentences can lead to increased ambiguity, potentially hindering the models' ability to focus on specific features.

Given these limitations we also experiment with the shortest sentences possible. Thus, we identify 8 new examples. In this context, each clause is pertinent to one, or at most two, instances of unfair contractual practices. A key advantage of this approach is that short sentences are simpler to process and reduce the risk of truncation. This approach may allow models to focus on specific patterns. However, it comes with potential issues. Indeed, there is a risk of oversimplification. Relying on short sentences may reduce cohesiveness and create a misalignment with real world contractual language, since ToS often contains long and complex sentences. Furthermore, models may lose the broader context necessary to fully understand the relationships between different elements. Thus, practices that depend on interconnected factors may not be fully captured.

By experimenting with the long and short sentence approaches, we aim to test their respective strengths and limitations to achieve the best possible outcomes in our few-shot settings.

5 Experimental Results

For our experimental evaluation, we consider the dataset described in Section 3, using the methodology illustrated in Section 4. Following

[1] At execution time, text within brackets is replaced with the corresponding entities.
[2] This line is not present in the zero-shot setting.

Table 2: Details of the generative models.

Model	#Params	Context Length	Quantization
Qwen [23]	32B	128k	4 bit
Codestral [51]	22B	32k	4 bit
Phi3 [53]	14B	128k	8 bit
Nemo [52]	12B	128k	8 bit
Llama [14]	8B	128k	8 bit
Mistral [25]	7B	32k	8 bit
Law-Chat [9]	7B	128k	8 bit

the setting of LexGLUE [8], we split the 142 documents into three sets: 85 documents for training, 35 for validation, and 22 for test. To address class imbalance, we evaluate all the approaches by measuring micro-F1 and macro-F1. As for LLMs, we consider 7 different models listed in Table 2: Llama3-8B, Mistral-7B, Law-Chat, MistralNemo-12B, Phi3-14B, Codestral-22B, Qwen-32B. All the experiments have been executed on two RTX 2080 Ti for a total of 22GB of VRAM. Quantization was employed when necessary, i.e., when the chosen model did not fit in the available VRAM. The 8-bit integer quantization consists of a mixed-precision decomposition where most of the values ($\approx$ 99.9%) are quantized in 8-bit precision, while outlier features are quantized to 16-bit [11]. Concerning 4-bit quantization, the QLoRA [12] method is employed.

5.1 Experimental Setting

All generative models are instruction-tuned LLMs between 7 and 32 billion parameters (Table 2). The input prompt for each model is pre-processed using the recommended template for that particular model. Specific tokens to the input prompt (e.g., [INST]) are appended and prepended, ensuring alignment with the model's native requirements. Temperature is set to 1, sampling is disabled, and the greedy search decoding method is employed to ensure reproducible results. The model weights are quantized to 8-bit or 4-bit integers, depending on the size of the model as indicated in Table 2, to meet the computational resources at our disposal.

We compare our generative approaches against several discriminative models: a linear SVM and 7 pre-trained transformers, as used in the LexGLUE benchmark [8]. The SVM is the same used in the original CLAUDETTE system [29]: it employs TF-IDF features using n-grams with $n \in [1, 2, 3]$, with hyperparameters optimized through grid search. All pre-trained models are fine-tuned for up to 20 epochs with early stopping on development data, using the Adam optimizer with a starting learning rate of $3e^{-5}$. Mixed precision (fp16) is used to reduce memory requirements, with a batch size of 8 across all experiments. Each setup is repeated five times with different random seeds, reporting test scores based on the seed yielding the best test performance.

5.2 Baselines

Table 3 shows the results obtained by baseline approaches, which are those used also within the LexGLUE benchmark [8]. Results are similar to those reported in LexGLUE, although we are hereby using a larger dataset of 142 documents, which are more diverse in terms of market sectors, and thus more challenging. Furthermore, the privacy included category utilized in this dataset was not part of the LexGLUE benchmark. Custom LegalBERT is the best performing approach, with 95.9/77.9 of micro/macro-F1, respectively.

5.3 Comparison of prompt strategies

Table 4 shows the results obtained with the three different prompt strategies described in Section 4. When using few-shot learning, we also report the results obtained with both short and long examples.

It is straightforward to observe that the performance of LLMs is significantly lower than those obtained by the baselines. The *pipeline* strategy is the one that consistently performs better than the others, for all the considered LLMs. This is probably due to the fact that LLMs, as observed in our preliminary experiments, tend to consider unfair also clauses that are totally unrelated to any unfairness category. Therefore, splitting the task into two subsequent stages, and first filtering out the clauses that are unrelated to unfairness categories, greatly simplifies the problem for LLMs (each of the two prompts is much simpler to address).

Regarding the complexity of examples provided in the few-shot setting, there is no huge difference in adopting short or long examples, with a slight advantage in using the latter. While small-size LLMs do not produce satisfying results, it is evident that performance increases with model size: Qwen-32B, although quantized with 4 bits only, achieves results that are not far from the best-performing BERT-based baselines. On the other hand, Codestral-22B is underperforming even smaller LLMs. These suboptimal performances can be attributed to its pretraining on a corpus predominantly composed of code, which significantly diverges from the linguistic and conceptual characteristics of legal texts required for the task at hand.

5.4 Computational Cost

Deberta-base and custom-legalbert require, respectively, 25 and 17 minutes for training on our architecture. For what concerns inference, both require between 1 and 2 milliseconds for the complete classification of a clause. For LLMs, the inference time depends on the considered setting, with a minimum of about 1 second for each model, and a maximum that depends on the specific model: about 5 seconds for Mistral-7B and Phi3-14B, 4 for Nemo-12B, and 3 for Llama3-8B. For all models, the single-promp approach is the fastest and the multi-prompt few shot long is the slowest. While training transformers clearly increases the computational cost in terms of training time, their inference time is about 1000 times faster than LLMs. Therefore, they seem a better alternative than LLMs both in terms of F1 score and inference time. However, it is important to remember that they are a viable solution only in scenarios where training data are available and it is possible to afford the training cost. The size of the tested LLM does not seem to impact inference time strongly. On the contrary, the prompt approach and the length of the prompt can increase inference time up to 4 times.

5.5 Evaluation of models for each category of unfairness

Table 5 presents the F1 scores of the best models across all the unfairness categories. In particular, we compare transformer-based models, i.e., Legalbert and Deberta-base, with small (Phi3-14B, Llama3-8B, Law-Chat-7B, Mistral-7B, Nemo-12B) and large LLMs

Table 3: Evaluation of baseline methods on the validation and the test split. For validation, we report the average score over 5 training and the variance. For test, we report the result of the best model.

Model	Validation		Test			
	Micro-F1	Macro-F1	Micro-F1	Macro-F1	Macro Acc.	Macro Rec.
Bert-base-uncased	95.8 ± 0.1	74.8 ± 0.8	95.6	75.3	80.5	71.2
Roberta-base	94.6 ± 1.5	49.8 ± 25.7	95.6	70.2	72.4	68.8
Deberta-base	95.8 ± 0.1	73.7 ± 0.2	**95.8**	**77.8**	**82.5**	74.6
Longformer-base-4096	93.5 ± 1.5	32.4 ± 20.7	95.5	62.7	70.3	57.6
Bigbird-Roberta-base	95.7 ± 0.0	72.6 ± 2.3	95.3	75.3	77.3	74.9
Legal-Bert-base-uncased	96.0 ± 0.1	76.0 ± 1.4	95.6	76.9	78.5	**77.1**
Custom-Legalbert	95.9 ± 0.1	75.4 ± 0.3	**95.9**	**77.9**	81.2	75.6
TFIDF+SVM	95.0 ± 0.0	62.0 ± 0.0	94.9	61.0	80.7	52.4

(Codestral 22B and Qwen 32B). For each model, we consider only the setting that yields the best result.

The fine-tuned transformer-based models, Legalbert and Deberta-base, outperform other techniques across most categories, thus highlighting the benefits of tailoring models to legal datasets and corpora. As for the aggregate results presented in Table 4, small LLMs perform significantly worse both than large LLMs and than BERT-based approaches.

In particular, Legalbert achieves the highest F1 scores for unilateral change (0.78 vs 0.76), jurisdiction (0.96 vs 0.90), and unilateral termination (0.75 vs 0.73). Conversely, Deberta-base surpasses Legalbert in arbitration (0.67 vs. 0.56), content removal (0.70 vs 0.67), limitation of liability (0.71 vs 0.70), and contract by using clauses (0.82 vs. 0.79). The two models achieve an equal F1 score for choice of law (0.93).

Even the per-category results confirm that the pipeline strategy is the one achieving the best results. As noted above, Nemo-12B is the best performing LLM among the small-sized ones, in particular as regard to content removal (0.64), choice of law (0.80) and privacy included (0.45) for pipeline short. Similarly, Llama3-8B reports the best results for arbitration (0.49), unilateral change (0.64) and limitation of liability (0.55) for pipeline long. Phi3-14B outperforms in assessing jurisdiction (0.74) and unilateral termination clauses (0.61) in pipeline short.

The full comparison between the short and long examples confirms the observation made for aggregate results (Table 4), i.e., that that the selection of examples does not strongly influence the performance of small LLMs within the pipeline approach. Differently, LawChat-7B achieves the best results under the *Multi-prompt* short approach as regards consent by using (0.62). The best performing strategy for Mistral-7B is using *Pipeline-long*, but its results are significantly worse than those of other small LLMs.

The frequency and unfairness distribution of clauses in the dataset correlate with the performance trends of small-size LLMs. Categories with higher clause frequencies, such as Limitation of Liability and Unilateral Termination, generally see better performance from fine-tuned models, reflecting their ability to adapt to well-represented clause types. Conversely, less frequent categories like Privacy Included and Arbitration highlight the limitations of few-shot and pipeline methods and the need for extensive domain-specific training. Another factor that might have contributed to the poor performance of small LLMs is the complexity of some categories. For instance, while there is a similar number of arbitration and privacy included clauses in ToS, the latter are less uniform and can often be expressed spanning over several sentences, while arbitration clauses tend to have less variation in their phrasing. To make it clearer, consider the following examples.

> *Please review our Privacy Policy, which governs the use of personal information on the Site and to which Subscriber agrees to be bound as a user of the Site.* - 9GAG ToS

> *By using our Service, you agree to be bound by Section I of these Terms (General Terms"), which contains provisions applicable to all users of our Service, including visitors to the DeviantArt website (the "Site") [...] The terms of DeviantArt's privacy policy are incorporated into, and form a part of, these Terms.* - DeviantArt ToS

In the first case, the unfairness evaluation can be done by analyzing the clause in isolation, as its impact and implications are contained within a sentence. The clause directly binds the user to the privacy policy through a single, explicit statement. This standalone nature makes it easier the identification and classification of the clause. Conversely, in the second case, the unfairness arises from the interplay between two distinct clauses. By merely using the service, the consumer implicitly agrees not only to the ToS but also to the data processing terms outlined in the privacy policy. This layered structure can make it harder for small-size models, with length constraints and difficulty in capturing long-range dependencies, to correctly assess such clauses.

Finally, we note that small-size LLMs classify descriptive sentences as unfair. While related to a certain clause category, these sentences merely describe procedures or other details irrelevant to the fairness assessment. For instance, this is the case of arbitration clauses. In this context, small LLms often classify as unfair statements outlining arbitration procedures or formalities. Consider the following example taken from the Headspace ToS.

> *(b) Arbitration rules. The federal arbitration act governs the interpretation and enforcement of this dispute-resolution provision. Arbitration will be initiated through the American arbitration association ("AAA").*

Such misclassifications suggest that smaller models might lack the contextual understanding needed to distinguish between procedural descriptions and substantive clauses with potential unfairness implications.

Table 4: Evaluation of LLMs over the test set. For each column, we report the best result in bold.

Model	Technique	Examples	Metrics			
			Micro-F1	Macro-F1	Macro Acc.	Macro Rec.
Llama3-8B	Single-prompt	/	0.71	0.37	0.38	0.61
Mistral-7B	Single-prompt	/	0.55	0.23	0.25	0.42
Law-Chat	Single-prompt	/	0.22	0.09	0.17	0.07
Nemo-12B	Single-prompt	/	0.72	0.39	0.42	0.57
Phi3-14B	Single-prompt	/	0.60	0.33	0.30	0.58
Codestral-22B	Single-prompt	/	0.15	0.19	0.37	0.17
Qwen-32B	Single-prompt	/	0.87	0.49	0.43	0.64
LLama3-8B	Multi-prompt - Zero shot	/	0.88	0.47	0.43	0.57
Mistral-7B	Multi-prompt - Zero shot	/	0.41	0.16	0.14	0.80
Law-Chat	Multi-prompt - Zero shot	/	0.83	0.39	0.33	0.62
Nemo-12B	Multi-prompt - Zero shot	/	0.61	0.26	0.20	0.81
Phi3-14B	Multi-prompt - Zero shot	/	0.79	0.41	0.31	0.77
Codestral-22B	Multi-prompt - Zero shot	/	0.34	0.21	0.19	0.86
Qwen-32B	Multi-prompt - Zero shot	/	0.88	0.61	0.56	0.75
Llama3-8B	Multi-prompt - Few shot	Long	0.79	0.40	0.30	0.84
		Short	0.78	0.40	0.31	0.78
Mistral-7B	Multi-prompt - Few shot	Long	0.53	0.24	0.20	0.85
		Short	0.52	0.22	0.18	0.87
Law-Chat	Multi-prompt - Few shot	Long	0.51	0.31	0.26	0.71
		Short	0.71	0.43	0.41	0.62
Nemo-12B	Multi-prompt - Few shot	Long	0.32	0.15	0.15	0.86
		Short	0.31	0.14	0.14	0.87
Phi3-14B	Multi-prompt - Few shot	Long	0.72	0.43	0.34	0.85
		Short	0.80	0.52	0.40	0.80
Codestral-22B	Multi-prompt - Few shot	Long	0.41	0.24	0.21	0.87
		Short	0.44	0.26	0.21	**0.89**
Qwen-32B	Multi-prompt - Few shot	Long	0.90	0.59	0.47	0.84
		Short	0.88	0.61	0.51	0.85
Llama3-8B	Pipeline	Long	0.90	0.58	0.51	0.71
		Short	0.91	0.55	0.54	0.61
Mistral-7B	Pipeline	Long	0.74	0.29	0.24	0.70
		Short	0.59	0.24	0.19	0.87
Law-Chat	Pipeline	Long	0.43	0.20	0.17	0.87
		Short	0.46	0.22	0.18	0.85
Nemo-12B	Pipeline	Long	0.92	0.62	0.57	0.72
		Short	0.94	0.62	0.71	0.59
Phi3-14B	Pipeline	Long	0.87	0.56	0.44	0.84
		Short	0.87	0.57	0.46	0.82
Codestral-22B	Pipeline	Long	0.80	0.45	0.35	0.85
		Short	0.80	0.47	0.36	0.85
Qwen-32B	Pipeline	Long	**0.95**	**0.72**	**0.77**	0.71
		Short	0.94	0.71	**0.77**	0.71

Table 5: Evaluation of models for each category of unfairness. For each column, we report the best results in bold and underline the best result of small LLMs.

Model	Technique	Category F1									
		fair	a	ch	cr	j	law	ltd	ter	use	pinc
Baselines											
Custom-Legalbert	Fine tuned	0.97	0.56	**0.78**	0.67	**0.96**	**0.93**	0.70	**0.75**	0.79	0.67
Deberta-base	Fine tuned	**0.98**	**0.67**	0.76	**0.70**	0.90	**0.93**	**0.71**	0.73	**0.82**	0.57
Small LLMs											
Mistral-7B	Pipeline-Long	0.85	0.07	0.19	0.08	0.12	0.1	0.3	0.38	0.14	0.20
LawChat	Multi Few-Short	0.87	0.36	0.32	0.08	0.65	0.24	0.19	0.55	0.62	0.38
Phi3-14B	Pipeline-Short	0.93	0.33	0.52	0.45	0.74	0.71	0.46	0.61	0.55	0.44
Llama3-8B	Pipeline-Long	0.95	0.49	0.64	0.55	0.68	0.54	0.55	0.58	0.42	0.36
Nemo-12B	Pipeline-Short	0.97	0.34	0.63	0.64	0.73	0.80	0.54	0.55	0.56	0.45
Large LLMs											
Qwen-32B	Pipeline-Long	**0.98**	0.48	0.70	0.64	0.91	**0.93**	0.67	0.57	0.65	**0.69**
Codestral-22B	Pipeline-Short	0.90	0.19	0.54	0.37	0.53	0.47	0.54	0.47	0.18	0.47

Overall, none of the small LLMs consistently outperform the others and all fall short as compared to transformer-based models. Of the two larger models, only Qwen-32b obtains comparable or better results than fine-tuned models in a few categories (e.g., choice of law and privacy included).

6 Conclusion

In this paper, we evaluate the capabilities of LLMs compared to BERT-based models in detecting unfair clauses in ToS.

Identifying the optimal prompt for an LLM on a task is a challenging problem, since LLMs are sensitive to context and phrasing in ways that are often unpredictable and difficult to fully explain [16, 31, 35, 46, 48, 57, 58]. Thus, we test several prompting strategies across five small LLMs, and two larger LLMs, against the state-of-the-art transformer models. Small LLMs could provide an attractive alternative to the supervised systems, including reduced dependency on extensive training data, lower resource requirements, and a smaller environmental footprint compared to larger models.

Despite these benefits, our experimental results show that small LLMs currently fall short of delivering satisfactory performance in unfair clause detection. Among small models, Nemo-12B emerges as the best-performing one, with a pipeline approach that decomposes the task into category identification and unfairness assessment, yielding the best results. However, no significant differences are observed between short and long example prompts. A larger model, Qwen-32B, demonstrates promising performance, with potential for further enhancement through fine-tuning, suggesting that model size may play a key role. Conversely, Codestral-22B shows poor results across the board. Based on these findings, we conclude that, at this stage, BERT-based models remain superior for detecting and classifying unfair clauses in consumer contracts.

In conclusion, our experimental results show that small LLMs face significant limitations in processing the complex and varied language structures of legal contracts. In particular, we identify three main issues regarding: (a) contextual understanding capacities; (b) domain and task-specific adaptation; (c) generalization and flexibility. Indeed, small models seem to focus on basic patterns, flagging irrelevant clauses as unfair or missing relevant ones. As noted in Section 5, they tend to consider as unfair descriptive sentences as well as clauses completely unrelated to any unfairness category. They show a restricted ability to grasp and fully understand the context needed to distinguish between procedural descriptions and substantive clauses with unfairness implications. Moreover, without the benefit of fine-tuning, small LLMs seem to struggle when encountering unfamiliar domain-specific legal terminology or varied phrasings.

More generally, their narrow language understanding can be a significant drawback in the analysis of legal contracts. Indeed, small language models seem unable to generalize concepts, even when provided with relevant examples.

In our future work, we plan to test parameter-efficient fine-tuning procedures, such as LoRA [22], to fine-tune LLMs on our data or other legal corpora. We will study the impact of retrieval-augmented generation architectures, exploiting legal rationales as an external knowledge base. We will also consider using LLMs to assist in the creation of a training corpus (e.g., following a recent study [32]). Another line of research would address the performance of LLMs in a multilingual context, such as CLAUDETTE's multilingual corpora [13, 15]. Finally, the automatic refinement of guidelines and annotations to address potential changes in the legislation would be a challenging topic. Possible solutions may rely on memory networks [45] and RAG-based solutions.

Acknowledgments

This work was partially supported by the following projects: CompuLaw – Computable Law – funded by the ERC under the Horizon 2020 (Grant Agreement N. 833647); PRIN2022 PRIMA - PRivacy Infringements Machine-Advice (Ref. Prot. n.: 20224TPEYC - CUP J53D23005130001); PRIN2022 EQUAL – EQUitableALgorithms (Ref. Prot n. 2022KFLF3E_001 - CUP J53D23005560001); CLAUDETTE IV, funded by the EUI Research Council for funding; "FAIR - Future Artificial Intelligence Research" – Spoke 8 and Spoke 1 (CAI4DSA

action) under the European Commission's NextGeneration EU programme, PNRR – M4C2 – Inv. 1.3, Partenariato Esteso (PE00000013); "Consumer Law and the Attention Economy", funded by National Science Centre, Poland (project no. 2022/45/B/HS5/01419).

References

[1] Orlando Amaral, Sallam Abualhaija, Damiano Torre, Mehrdad Sabetzadeh, and Lionel C Briand. 2021. AI-enabled automation for completeness checking of privacy policies. *IEEE Trans. on Software Engineering* 48, 11 (2021), 4647–4674.

[2] Quentin André, Ziv Carmon, Klaus Wertenbroch, Alia Crum, Douglas Frank, William Goldstein, Joel Huber, Leaf Van Boven, Bernd Weber, and Haiyang Yang. 2018. Consumer choice and autonomy in the age of artificial intelligence and big data. *Customer Needs and Solutions* 5, 1 (2018), 28–37.

[3] Yannis Bakos, Florencia Marotta-Wurgler, and David R Trossen. 2014. Does anyone read the fine print? Consumer attention to standard-form contracts. *The Journal of Legal Studies* 43, 1 (2014), 1–35.

[4] Omri Ben-Shahar. 2013. Regulation through boilerplate: an apologia. *Mich. L. Rev.* 112 (2013), 883.

[5] David Bernhard, Luka Nenadic, Stefan Bechtold, and Karel Kubicek. 2025. Multilingual Scraper of Privacy Policies and Terms of Service. In *CS and LAW 2025*. Association for Computing Machinery, 55–63.

[6] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems* 33 (2020), 1877–1901.

[7] Ilias Chalkidis. 2023. ChatGPT may Pass the Bar Exam soon, but has a Long Way to Go for the LexGLUE benchmark.

[8] Ilias Chalkidis, Abhik Jana, Dirk Hartung, Michael Bommarito, Ion Androutsopoulos, Daniel Katz, and Nikolaos Aletras. 2022. LexGLUE: A Benchmark Dataset for Legal Language Understanding in English. In *ACL*. 4310–4330.

[9] Daixuan Cheng, Shaohan Huang, and Furu Wei. 2024. Adapting Large Language Models via Reading Comprehension. In *ICLR*. OpenReview.net.

[10] Jose M Del Alamo, Danny S Guaman, Boni García, and Ana Diez. 2022. A systematic mapping study on automated analysis of privacy policies. *Computing* 104, 9 (2022), 2053–2076.

[11] Tim Dettmers, Mike Lewis, Younes Belkada, and Luke Zettlemoyer. 2022. GPT3.int8(): 8-bit Matrix Multiplication for Transformers at Scale. In *NeurIPS*.

[12] Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. 2023. QLoRA: Efficient Finetuning of Quantized LLMs. In *NeurIPS*.

[13] Kasper Drawzeski, Andrea Galassi, Agnieszka Jablonowska, Francesca Lagioia, Marco Lippi, Hans Wolfgang Micklitz, Giovanni Sartor, Giacomo Tagiuri, and Paolo Torroni. 2021. A Corpus for Multilingual Analysis of Online Terms of Service. In *NLLP*. 1–8. doi:10.18653/v1/2021.nllp-1.1

[14] Abhimanyu Dubey and Abhinav Jauhri et al. 2024. The Llama 3 Herd of Models.

[15] Andrea Galassi, Francesca Lagioia, Agnieszka Jabłonowska, and Marco Lippi. 2024. Unfair clause detection in terms of service across multiple languages. *Artificial Intelligence and Law* (2024), 1–49. doi:10.1007/s10506-024-09398-7

[16] Chengguang Gan and Tatsunori Mori. 2023. Sensitivity and Robustness of Large Language Models to Prompt Template in Japanese Text Classification Tasks. In *PACLIC*. Association for Computational Linguistics, 1–11.

[17] Sofia Grafanaki. 2016. Autonomy challenges in the age of big data. *Fordham Intell. Prop. Media & Ent. LJ* 27 (2016), 803.

[18] Alfonso Guarino, Nicola Lettieri, Delfina Malandrino, and Rocco Zaccagnino. 2021. A machine learning-based approach to identify unlawful practices in online terms of service: analysis, implementation and evaluation. *Neural Computing and Applications* 33 (2021), 17569–17587.

[19] Neel Guha, Julian Nyarko, Daniel E. Ho, Christopher Ré, Adam Chilton, K. Aditya, Alex Chohlas-Wood, Austin Peters, Brandon Waldon, Daniel N. Rockmore, Diego Zambrano, et al. 2023. LegalBench: A Collaboratively Built Benchmark for Measuring Legal Reasoning in Large Language Models. In *NeurIPS*.

[20] Hamza Harkous, Kassem Fawaz, Rémi Lebret, Florian Schaub, Kang G. Shin, and Karl Aberer. 2018. Polisis: Automated Analysis and Presentation of Privacy Policies Using Deep Learning. In *USENIX Security Symposium*. 531–548.

[21] Natali Helberger, Orla Lynskey, Hans-W. Micklitz, Peter Rott, Marijn Sax, and Joanna Strycharz. 2021. *EU Consumer Protection 2.0: Structural Asymmetries in Digital Consumer Markets.* Technical Report.

[22] Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. LoRA: Low-Rank Adaptation of Large Language Models. In *ICLR*. OpenReview.net.

[23] Binyuan Hui, Jian Yang, Zeyu Cui, Jiaxi Yang, Dayiheng Liu, Lei Zhang, Tianyu Liu, Jiajun Zhang, Bowen Yu, Kai Dang, An Yang, Rui Men, Fei Huang, Xingzhang Ren, Xuancheng Ren, Jingren Zhou, and Junyang Lin. 2024. Qwen2.5-Coder Technical Report.

[24] Agnieszka Jablonowska, Francesca Lagioia, Marco Lippi, Hans-Wolfgang Micklitz, Giovanni Sartor, and Giacomo Tagiuri. 2021. Assessing the Cross-Market Generalization Capability of the CLAUDETTE System. In *JURIX*. 62–67.

[25] Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2023. Mistral 7B.

[26] Daniel Jurafsky and James H. Martin. 2000. Speech and language processing.

[27] Francesca Lagioia, Agnieszka Jabłonowska, Ruta Liepina, and Kasper Drazewski. 2022. AI in search of unfairness in consumer contracts: the terms of service landscape. *Journal of Consumer Policy* 45, 3 (2022), 481–536.

[28] Marco Lippi, Giuseppe Contissa, Agnieszka Jablonowska, Francesca Lagioia, Hans-Wolfgang Micklitz, Przemyslaw Palka, Giovanni Sartor, and Paolo Torroni. 2020. The force awakens: Artificial intelligence for consumer law. *Journal of artificial intelligence research* 67 (2020), 169–190.

[29] Marco Lippi, Przemyslaw Palka, Giuseppe Contissa, Francesca Lagioia, Hans-Wolfgang Micklitz, Giovanni Sartor, and Paolo Torroni. 2019. CLAUDETTE: an automated detector of potentially unfair clauses in online terms of service. *Artif. Intell. Law* 27, 2 (2019), 117–139. doi:10.1007/S10506-019-09243-2

[30] Marco Loos and Joasia Luzak. 2016. Wanted: a bigger stick. On unfair terms in consumer contracts with online service providers. *Journal of consumer policy* 39 (2016), 63–90.

[31] Yao Lu, Max Bartolo, Alastair Moore, Sebastian Riedel, and Pontus Stenetorp. 2022. Fantastically Ordered Prompts and Where to Find Them: Overcoming Few-Shot Prompt Order Sensitivity. In *ACL (1)*. Association for Computational Linguistics, 8086–8098.

[32] Yuxuan Lu, Bingsheng Yao, Shao Zhang, Yun Wang, Peng Zhang, Tun Lu, Toby Jia-Jun Li, and Dakuo Wang. 2023. Human Still Wins over LLM: An Empirical Study of Active Learning on Domain-Specific Annotation Tasks. *CoRR* abs/2311.09825 (2023).

[33] Alexandra Sasha Luccioni, Sylvain Viguier, and Anne-Laure Ligozat. 2023. Estimating the carbon footprint of bloom, a 176b parameter language model. *Journal of Machine Learning Research* 24, 253 (2023), 1–15.

[34] Thomas J Maronick. 2014. Do consumers read terms of service agreements when installing software?-a two-study empirical analysis. *International Journal of Business and Social Research* 4, 6 (2014), 137–145.

[35] Eric Martínez. 2024. Re-evaluating GPT-4's bar exam performance. *Artificial Intelligence and Law* (2024).

[36] Hans-W Micklitz, Przemysław Pałka, and Yannis Panagis. 2017. The empire strikes back: digital control of unfair terms of online services. *Journal of consumer policy* 40 (2017), 367–388.

[37] Eliza Mik. 2016. The erosion of autonomy in online consumer transactions. *Law, Innovation and Technology* 8, 1 (2016), 1–38.

[38] Joel Niklaus, Veton Matoshi, Pooja Rani, Andrea Galassi, Matthias Stürmer, and Ilias Chalkidis. 2023. LEXTREME: A Multi-Lingual and Multi-Task Benchmark for the Legal Domain. In *Findings of the Association for Computational Linguistics: EMNLP 2023*. Association for Computational Linguistics, Singapore, 3016–3054. doi:10.18653/v1/2023.findings-emnlp.200

[39] Jonathan A Obar and Anne Oeldorf-Hirsch. 2020. The biggest lie on the internet: Ignoring the privacy policies and terms of service policies of social networking services. *Information, Communication & Society* 23, 1 (2020), 128–147.

[40] Przemyslaw Palka. 2023. Terms of injustice. *W. Va. L. Rev.* 126 (2023), 133.

[41] Przemysław Pałka, Radosław Pałosz, Andrzej Porębski, and Katarzyna Wiśniewska. 2024. A dataset on the contents of 100 terms of service of online platforms, analyzed and evaluated under the EU consumer law. *Data in Brief* 53 (2024), 110136.

[42] Ali Hakimi Parizi, Yuyang Liu, Prudhvi Nokku, Sina Gholamian, and David Emerson. 2023. A Comparative Study of Prompting Strategies for Legal Text Classification. In *NLLP*. 258–265.

[43] Christine Riefa and Mateusz Grochowski. 2023. The enforcement of EU consumer law. In *Research Handbook on the Enforcement of EU Law*. 350–364.

[44] David Rodriguez, Ian Yang, Jose M Del Alamo, and Norman Sadeh. 2024. Large language models: a new approach for privacy policy analysis at scale. *Computing* 106, 12 (2024), 3879–3903.

[45] Federico Ruggeri, Francesca Lagioia, Marco Lippi, and Paolo Torroni. 2022. Detecting and explaining unfairness in consumer contracts through memory networks. *Artificial Intelligence and Law* 30, 1 (2022), 59–92.

[46] Abel Salinas and Fred Morstatter. 2024. The Butterfly Effect of Altering Prompts: How Small Changes and Jailbreaks Affect Large Language Model Performance. In *ACL (Findings)*. Association for Computational Linguistics, 4629–4651.

[47] Timo Schick and Hinrich Schütze. 2021. It's Not Just Size That Matters: Small Language Models Are Also Few-Shot Learners. In *NAACL-HLT*. Association for Computational Linguistics, 2339–2352.

[48] Melanie Sclar, Yejin Choi, Yulia Tsvetkov, and Alane Suhr. 2024. Quantifying Language Models' Sensitivity to Spurious Features in Prompt Design or: How I learned to start worrying about prompt formatting. In *ICLR*. OpenReview.net.

[49] Jinzhe Tan, Hannes Westermann, and Karim Benyekhlef. 2023. ChatGPT as an Artificial Lawyer?. In *AI4AJ@ICAIL*, Vol. 3435. CEUR-WS.org.

[50] Chenhao Tang, Zhengliang Liu, Chong Ma, Zihao Wu, Yiwei Li, Wei Liu, Dajiang Zhu, Quanzheng Li, Xiang Li, Tianming Liu, and Lei Fan. 2023. PolicyGPT: Automated Analysis of Privacy Policies with Large Language Models. *CoRR* abs/2309.10238 (2023).
[51] Mistral AI team. 2024. Codestral. https://mistral.ai/news/codestral/
[52] Mistral AI team. 2024. Mistral nemo. https://mistral.ai/news/mistral-nemo/
[53] Microsoft Phi-3 team. 2024. Phi-3 Technical Report: A Highly Capable Language Model Locally on Your Phone.
[54] Welderufael B. Tesfay, Peter Hofmann, Toru Nakamura, Shinsaku Kiyomoto, and Jetzabel M. Serna. 2018. PrivacyGuide: Towards an Implementation of the EU GDPR on Internet Privacy Policy Evaluation. In *IWSPA@CODASPY*. ACM, 15–21.
[55] Elisa Tsai, Neal Mangaokar, Boyuan Zheng, Haizhong Zheng, and Atul Prakash. 2025. Harmful Terms and Where to Find Them: Measuring and Modeling Unfavorable Financial Terms and Conditions in Shopping Websites at Scale. In *WWW*. ACM, 990–1003.
[56] Fangyi Yu, Lee Quartey, and Frank Schilder. 2022. Legal Prompting: Teaching a Language Model to Think Like a Lawyer. *CoRR* abs/2212.01326 (2022).
[57] Zihao Zhao, Eric Wallace, Shi Feng, Dan Klein, and Sameer Singh. 2021. Calibrate before use: Improving few-shot performance of language models. In *International conference on machine learning*. PMLR, 12697–12706.
[58] Chujie Zheng, Hao Zhou, Fandong Meng, Jie Zhou, and Minlie Huang. 2024. Large Language Models Are Not Robust Multiple Choice Selectors. In *ICLR*.