SaToS: Assessing and Summarising Terms of Services from German Webshops

Daniel Braun and **Elena Scepankova** and **Patrick Holl** and **Florian Matthes**
Technical University of Munich
Department of Informatics
{daniel.braun,elena.scepankova,patrick.holl,matthes}@tum.de

Abstract

Every time we buy something online, we are confronted with Terms of Services. However, only a few people actually read these terms, before accepting them, often to their disadvantage. In this paper, we present the SaToS browser plugin which summarises and simplifies Terms of Services from German webshops.

1 Introduction

The phrase "I have read and understood the terms of service" is often referred to as "the biggest lie on the internet" (Pridmore and Overocker, 2014; Binns and Matthews, 2014). In a study conducted by Obar and Oeldorf-Hirsch (2016), participants were asked to register for a made-up social network. 74% of the participants did not read the Terms of Service (ToS) at all and those who did read it spent on average 13.6 seconds on it, hardly enough to read let alone understand a juridical text with more than 4,300 words. Nevertheless, all participants agreed to the ToS.

General terms and conditions (German: *Allgemeine Geschäftsbedingungen - AGB*; in the following: ToS) included in standard form contracts are of significant economic value, as most companies use these terms when entering into contractual relationships with their customers. Historically, ToS trace back to the age of industrialisation in the 19th century. In the course of mass production, entering into contracts has been accompanied by the unilateral use of these terms and conditions as a set of pre-formulated rules - tailored to one party's own purposes and thus resulting in an imbalance of powers between the contracting parties. (Zerres, 2014)

In this paper, we present the ongoing interdisciplinary computer and legal science research project SaToS (Software aided analysis of ToS) and a prototype which automatically identifies ToS on German webshops and summarises them with regard to their lawfulness and customer friendliness, in a simplified language. These summarisations are presented through an adblocker-like browser plugin. In this way, SaToS aims to empower customers to make educated decisions about where to buy or not within seconds, directly addressing the imbalance of powers and fostering the constitutional principle of Legal Clarity[1].

2 Related Work

Generally speaking, automatically generated summarisations can be divided into extractive and abstractive (cf. e.g. Das and Martins (2007)). As mentioned before, many people do not read ToS at all and even if they do, these texts are often difficult to understand. Therefore, in order to make ToS understandable for customers, it is necessary to create abstractive, simplified summaries, rather than extractive ones. Currently, there are mainly two projects trying to create automatic summarisations of legal texts: the SUM project from Grover et al. (2003) and the LetSum project from Farzindar and Lapalme (2004). However, both systems create extractive summaries for English texts, while we aim to create abstractive summaries for German texts. In order to create abstractive summaries, a system first

[1] Art. 20 Abs. 3 GG (Grundgesetz - German Constitution)

Proceedings of The 10th International Natural Language Generation conference, pages 223–227,
Santiago de Compostela, Spain, September 4-7 2017.

has to obtain the relevant information from the text. Information Retrieval (IR) for legal texts has gained a lot of attraction in recent years. Examples are McCallum (2005), Grabmair et al. (2015), Francesconi et al. (2010), and Shulayeva et al. (2017), or, for German texts, Walter and Pinkal (2006), and Waltl et al. (2017). The issue of simplifying legal texts was e.g. addressed by Bhatia (1983) and Collantes et al. (2015). A general architecture for simplifying texts was presented by Siddharthan (2002). From a legal perspective, ToS;DR (Binns and Matthews, 2014) and janolaw[2] pursue a similar aim by evaluating ToS. Whereas we use a natural language processing and artificial intelligence in order to assess and evaluate ToS, they are crowed-sourced, which affects their scalability and topicality.

3 Legal Assessment

The assessment of ToS is of enormous value. Firstly, they affect many important issues of contractual relationships - details of performance and payment, liability, revocation rights, the place of jurisdiction - and thus have a significant impact on the customer. Being drafted unilaterally by one party, they bear risks like limiting liability or revocation rights, granting permission to increase prices or imposing penalties in case of delay etc.

Secondly, legal language is written in a way that is often difficult to understand. The reason is that law by itself has to be written with a certain degree of abstractness, in order to fulfil its function of regulating our social behaviour. This abstractness, however, leads to a low level of comprehensibility. Although law has to satisfy both principles - abstractness and comprehensibility -, it implicitly favours the former at costs of the latter. Against the background of this, it is not surprising that people avoid reading general terms and conditions at all.

Finally, the imbalance of powers resulting from the fact, that ToS are imposed unilaterally by one party, is mirrored by the number of laws[3] and courts decisions[4] assessing the lawfulness of those clauses.

[2]https://www.janolaw.de/

[3]Relevant laws concerning general terms and conditions are in §§305 -310 BGB (Bürgerliches Gesetzbuch - German Civil Code).

[4]According to the data base *Juris*, there are currently about 27,600 judgments addressing the lawfulness of ToS.

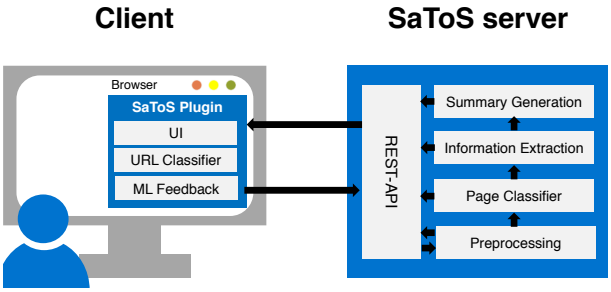

Figure 1: Architecture of the SaToS prototype

We try to adjust this imbalance of powers by identifying unlawful clauses and indicating differences between them so that customers will finally know their rights, without any previously required legal knowledge.

4 Prototype Architecture

The client-server architecture of the SaToS prototype is shown in Figure 1. While most of the natural language processing and generation is done on the server, the client handles the output and collects feedback from the user. A REST-API is used for the communication between client and server. The server itself is a Node.js application and internally based on a pipes and filters architecture (Meunier, 1995).

4.1 SaToS server

In this section, we will describe the components of the SaToS server. The REST-API consists of routes for every of these components. While it is possible to exit the pipeline after every module, it can only be entered through the first module. All routes take a URL as input.

4.1.1 Preprocessing

In the first module, the content of the webpage is retrieved and pre-processed. First, the main content is extracted and elements like navigation and header, are removed. Afterwards, all HTML tags are removed. Depending on further analysis that will be conducted, additional pre-processing is conducted, like tokenization, stemming, and POS tagging.

4.1.2 Page Classifier

The Page Classifier is a binary classifier that labels each page either “ToS” or “Other”. In our current prototype, we use a naive Bayes classifier. Our

aim is to incrementally improve its quality based on user feedback (cf. Section 4.2.1). In Section 5, we present an evaluation of the classifier.

4.1.3 Information Extraction

The Information Extraction (IE) module contains the domain knowledge, which is necessary to extract the information and identify unlawful clauses by examining the used legal language formulations, in accordance with the rules used by German courts. An excerpt of these rules is shown in Table 1.

There is not a single module for IE, but one for every aspect. Currently, our prototype has two of these modules, one for the right of withdrawal and one for the right of warranty. We decided to start with these, because they are very valuable for potential customers, included in most ToS, and relatively similar because they both essentially describe a timespan. Therefore, in order to extract this information, we first look for sentences which describe a timespan, i.e. a number or numeral word followed by a "unit" like day, month, or year. Afterwards, we identify the topic of the sentence. Thanks to the legal nature of the texts, there is a relatively small variety of permissible formulations to describe the right of withdrawal (*"Widerrufsrecht"*) and the warranty period (*"Gewährleistungsfrist"*). Once the information is extracted, it is returned in JSON-format. The sentence *"Der Kunde kann von uns erhaltene Ware ohne Angabe von Gründen innerhalb von 30 Tagen durch Rücksendung der Ware zurückgeben."*[5] would, for example, generate the output shown in Listing 1.

```
{
    "topic": "Widerrufsrecht",
    "dataType": "Timespan",
    "value": 30,
    "unit": "Tag"
}
```

Listing 1: Format of extracted Information

4.1.4 Summary Generation

The summary generator gets an array of extracted information in the above-described format as input and is leant on the architecture described by Reiter (2007). Since we do not have purely numerical data input, we do not have a *Signal Analysis* stage, however, the above-described information extraction fulfils a similar goal. The next stage is *Data Interpretation*. In this stage, we interpret the extracted information mainly regarding their legality. In this way, we distinguish between unlawful, lawful, and customer friendly regulations. Under German law, for example, customers always have to have at least 14 days of time to withdraw their order. Hence, a shorter timespan would be unlawful, a timespan of 14 days would be lawful, and anything beyond would be classified as customer friendly.

During *Document Planning*, so far, only the order of the messages is determined, starting with unlawful messages, followed by customer friendly messages, followed by lawful messages.

Finally, during *Realisation*, the actual summaries are created based on templates that have been written by a jurist. These templates are designed to be easily understandable while still containing all the necessary information and have to be created for each individual information extraction module.

4.2 Browser Plugin

The SaToS browser plugin works passively and does usually not require any user input.

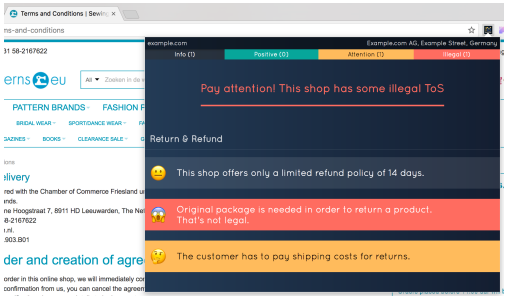

Figure 2: SaToS Browser Plugin

4.2.1 User Interface

Figure 2 shows the UI of the Plugin, including an overall recommendation for the shop and three categorised summary excerpts. The summary excerpts are split into the four categories: *Info* (neutral information for the user, grey), *Positive* (anything that improves customer friendliness and which goes beyond the legal requirements, green), *Attention* (warns the user if the ToS contains clauses which are legal but unusual, orange), and *Illegal* (any invalid or illegal

[5] https://www.thomann.de/de/compinfo_terms.html; last accessed 12 May 2017

	unlawfull	rules (translated from German)
Right of warranty	New goods: less than 2 years; used goods: less than 1 year	-warranty . . . ([0-9]* | [one | two | . . .]) [day(s) | month(s) | year(s)] AND used OR NOT used (goods | products) -warranty . . . used (goods | products) . . . excluded
Right of withdrawal	Products have to be send back using the original packaging	-product . . . original (packaging | packed | . . .) . . . (return | send back) -original (packaging | packed | . . .) . . . (return | send back)
Period for withdrawal	Period of less than 14 days for shops trading in the EU	-withdraw . . . ([0-9]* | [one | two | . . .]) [day(s) | month(s) | year(s)]
Obligation to inspect product	Warranty rights only if customer inspects and/or reports any product defects	-warranty . . . [inspect|report] AND NOT merchant
Risk of loss	In case of shipped sales the customer bears the risk of loss	[risk of loss | bearing the risk] . . . [shipped | carriage of goods] . . . consumer

Table 1: Extraction rules for ToS (excerpt)

statement, red). The summaries are assigned to a certain category and highlighted accordingly. Furthermore, based on the categorizations, we generate a recommendation for the shop as a whole. Users can also give feedback whether the ToS were classified correctly. We use this feedback to realise an online learning approach for our ML algorithms.

4.2.2 URL Classifier

Usually, when a user visits a webshop, he enters it via a specific landing or product page. However, for the summarisation, we have to process the content of the shops' ToS page. The goal of the URL classifier is to pre-select links that potentially lead to the ToS page and hence restrict the set of pages that have to be classified by the server. The classification is done by using a rule-based approach that matches common patterns for ToS links. One common pattern we identified is that the URL often contains *"AGB"*. The classifier separates URL strings into the following components: scheme specifier, network location part, path, query parameters. The path and query parameters are matched against a set of pre-defined, weighted rules. If the matches reach a certain threshold, we consider that a given URL points to a potential ToS page.

5 Evaluation ToS Classification

As mentioned before, we use a hybrid approach of client-based rules and server-based ML. Since all further analyses are based on the correct classification of ToS pages, we conducted an evaluation of both components. We collected a dataset of 3424 URLs. 2592 from ToS pages, manually labelled by a price comparison website, and 832 from other webshop pages. We split the dataset into training (200 ToS and 200 Other) and test (2392 ToS and 632 Other). The results of the evaluation are shown in Table 2. It is obvious, that the ML approach performed significantly better, with regard to precision, recall, and F-score. Given the fact, that the ML classifier was trained with a relatively small, non-optimized, dataset, the results are promising, keeping in mind that we use an online learning approach and expect the system to improve over time. One might wonder, why one should use a hybrid approach, although ML performs better in every category. The fifth column in Table 2 shows the average time in seconds, that was needed to classify a URL. If successful, the rule-based approach is not only faster, but its calculation is also "free" for SaToS, since it happens on the client.

approach	precision	recall	F-score	$\varnothing t$ in s
ML	0.9115	0.8219	0.8644	1.435
rule-based	0.7953	0.5393	0.6428	0.001

Table 2: Evaluation ToS Classification

6 Conclusion

By combining legal expertise with state-of-the-art technology, we want to empower customers to understand ToS and exercise their rights towards companies. In this paper, we presented a first research prototype, called SaToS, which automatically detects, summarises, and analyses ToS from German webshops regarding their lawfulness and customer-friendliness. We have evaluated the ToS Classifier and argued for a hybrid solution, combining rule-based approaches and ML. In the future, we want to expand the prototype for other ToS clauses.

References

Vijay K Bhatia. 1983. Simplification v. easification-the case of legal texts. *Applied linguistics*, 4:42.

Reuben Binns and David Matthews. 2014. Community structure for efficient information flow in'tos; dr', a social machine for parsing legalese. In *Proceedings of the 23rd International Conference on World Wide Web*, pages 881–884. ACM.

Miguel Collantes, Maureen Hipe, Juan Lorenzo Sorilla, Laurenz Tolentino, and Briane Samson. 2015. Simpatico: A text simplification system for senate and house bills. In *Proceedings of the 11th National Natural Language Processing Research Symposium*, pages 26–32.

Dipanjan Das and André FT Martins. 2007. A survey on automatic text summarization. *Literature Survey for the Language and Statistics II course at CMU*, 4:192–195.

Atefeh Farzindar and Guy Lapalme. 2004. Letsum, an automatic legal text summarizing system. *Legal knowledge and information systems, JURIX*, pages 11–18.

Enrico Francesconi, Simonetta Montemagni, Wim Peters, and Daniela Tiscornia. 2010. *Semantic processing of legal texts: Where the language of law meets the law of language*, volume 6036. Springer.

Matthias Grabmair, Kevin D Ashley, Ran Chen, Preethi Sureshkumar, Chen Wang, Eric Nyberg, and Vern R Walker. 2015. Introducing luima: an experiment in legal conceptual retrieval of vaccine injury decisions using a uima type system and tools. In *Proceedings of the 15th International Conference on Artificial Intelligence and Law*, pages 69–78. ACM.

Claire Grover, Ben Hachey, Ian Hughson, and Chris Korycinski. 2003. Automatic summarisation of legal documents. In *Proceedings of the 9th international conference on Artificial intelligence and law*, pages 243–251. ACM.

Andrew McCallum. 2005. Information extraction: Distilling structured data from unstructured text. *Queue*, 3(9):48–57.

Regine Meunier. 1995. The pipes and filters architecture. In *Pattern languages of program design*, pages 427–440. ACM Press/Addison-Wesley Publishing Co.

Jonathan A Obar and Anne Oeldorf-Hirsch. 2016. The biggest lie on the internet: Ignoring the privacy policies and terms of service policies of social networking services.

Jeannie Pridmore and John Overocker. 2014. Privacy in virtual worlds: a us perspective. *Journal For Virtual Worlds Research*, 7(1).

Ehud Reiter. 2007. An architecture for data-to-text systems. In *Proceedings of the Eleventh European Workshop on Natural Language Generation*, pages 97–104. Association for Computational Linguistics.

Olga Shulayeva, Advaith Siddharthan, and Adam Wyner. 2017. Recognizing cited facts and principles in legal judgements. *Artificial Intelligence and Law*, 25(1):107–126.

Advaith Siddharthan. 2002. An architecture for a text simplification system. In *Language Engineering Conference*, page 64. IEEE Computer Society.

Stephan Walter and Manfred Pinkal. 2006. Automatic extraction of definitions from german court decisions. In *Proceedings of the workshop on information extraction beyond the document*, pages 20–28. Association for Computational Linguistics.

B. Waltl, J. Landthaler, E. Scepankova, F. Matthes, T. Geiger, C. Stocker, and C. Schneider. 2017. Automated extraction of semantic information from german legal documents. In *IRIS: Internationales Rechtsinformatik Symposium*. Association for Computational Linguistics.

Thomas Zerres. 2014. Principles of the german law on standard terms of contract. *Jurawelt*.