Semantic Similarity based Clustering of License Excerpts for Improved End-User Interpretation

Najmeh Mousavi Nejad
University of Bonn / Fraunhofer IAIS
nejad@cs.uni-bonn.de

Simon Scerri
University of Bonn / Fraunhofer IAIS
scerri@cs.uni-bonn.de

Sören Auer
Leibniz Universität Hannover / TIB Information Center
soeren.auer@tib.eu

ABSTRACT

With the omnipresent availability and use of cloud services, software tools, Web portals or services, legal contracts in the form of End-User License Agreements (EULA) regulating their use are of paramount importance. Often the textual documents describing these regulations comprise many pages and can not be reasonably assumed to be read and understood by humans. In this work, we describe a method for extracting and clustering relevant parts of such documents, including permissions, obligations, and prohibitions. The clustering is based on semantic similarity employing a distributional semantics approach on large word embeddings database. An evaluation shows that it can significantly improve human comprehension and that improved feature-based clustering has a potential to further reduce the time required for EULA digestion. Our implementation is available as a web service, which can directly be used to process and prepare legal usage contracts.

CCS CONCEPTS

• **Information systems** → **Information extraction**; **Clustering and classification**; *Ontologies*; *Summarization*; • **Computing methodologies** → **Natural language processing**;

KEYWORDS

End-User License Agreements; EULA; Semantic Similarity; Clustering; Distributional Approach; Word Embeddings

ACM Reference format:
Najmeh Mousavi Nejad, Simon Scerri, and Sören Auer. 2017. Semantic Similarity based Clustering of License Excerpts for Improved End-User Interpretation. In *Proceedings of Semantics2017, Amsterdam, Netherlands, September 11–14, 2017,* 8 pages.
https://doi.org/10.1145/3132218.3132224

1 INTRODUCTION

The ever increasing use of online services, Web portals and software tools also resulted in a proliferation of extensive legal documents governing their use. Users tend to ignore these documents due to the time required for reading and the cumbersome legal lingua. However, it is crucial to be aware of the policies associated with online services and software tools. Each service has a set of terms and conditions, often called *End-User License Agreement* (EULA), which should be agreed by users, prior to their usage of that service. Unsurprisingly, based on the recent surveys, it is very common for people to ignore EULAs. According to an online research commissioned by *Skandia*, only 7% read online EULAs when signing up for products and services[1]. Of those who were surveyed, 21% said that they felt uncomfortable as a result of ticking an EULA acceptance box without reading the document: 10% found themselves locked into a longer term contract than they expected and 5% lost money by not being able to cancel or amend subscriptions and bookings.

In order to ease the process of license analysis for users, we can apply text mining techniques to extract important segments and sentences from EULAs and provide a compressed and summarized version to end-users. We introduce a novel approach that exploits the semantic similarity between short text excerpts to cluster similar policies in EULAs. In previous phase of our project, the important segments and sentences were extracted from the natural language licenses and were categorized into three important classes (e.g., `permission`, `prohibition` and `duty`) [19]. During our experiments, we have noticed that some sentences in the classes have similar or close meaning to each other and therefore can be categorized together. Table 1 shows similar extracted `duties` from the Apache License[2]. The colored words in this table have close or similar meanings and therefore the segments can be grouped together.

Table 1: Example of Similar Extracted Duties

You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works.
If the Work includes a NOTICE text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works.

It should be clarified here that our approach does not intend to remove any extracted segment from similar ones, since this may lead to losing vital information in EULAs. Instead our goal is to provide a brief summary for each cluster. If the end-user is

Semantics2017, September 11–14, 2017, Amsterdam, Netherlands

ACM ISBN 978-1-4503-5296-3/17/09…$15.00
https://doi.org/10.1145/3132218.3132224

[1] http://www.prnewswire.co.uk/news-releases/skandia-takes-the-terminal-out-of-terms%2Dand%2Dconditions-145280565.html
[2] https://www.apache.org/licenses/LICENSE-2.0

concerned about a specific policy, they can browse the list of items in each cluster and see the details.

In this study, we have taken `permissions`, `prohibitions` and `duties` excerpts as the input and extracted key features from them in order to perform clustering. After feature extraction, a semantic similarity framework based on word embeddings is used to compute similarity between different features of each class and then by summation of features similarities, the final similarity score for each pair is calculated and a symmetric similarity matrix is built for each class. Finally, the most similar segments in each class are grouped together with a hierarchical agglomerative clustering (HAC) algorithm and the procedure goes on until a certain threshold is reached. Our contributions are in particular:

- Identification of crucial features with JAPE grammar rules [6], specifically tailored for EULAs.
- Creation of a large domain-specific word embedding database with 1,000 EULAs.
- Implementation of a license analysis framework integrating different APIs including GATE [7], Stanford CoreNLP [17] and DISCO [13], which comprises a web API and UI.
- A comprehensive evaluation of our approach benchmarking it against human judgment.

The rest of the paper is organized as follows: we provided a short literature review in section 2; in section 3 we explain our approach and its implementation in the *EULAide* Service; section 4 evaluates the clustering alternatives as well as the usability of *EULAide* service; and section 5 concludes with a list of possible directions for future work.

2 RELATED WORK

An in-depth literature survey revealed that there is no automated approach for the required EULA interpretation and summarization to support end-users with their comprehension. Previous related efforts have focussed on general document clustering and summarization, and did not consider the legal references and characteristics of EULAs. The only comparable service assisting users to understand common licenses is `tldrlegal`[3], which is based on crowdsourcing strategy and is manually supported by experts in the field. Our implemented methods in *EULAide* attempt to eliminate the dependency on human involvement.

Over the years, many clustering methods have been proposed for the summarisation of texts. Survey papers such as [1] indicate a large variety of methods can be pursued. Efforts investigating semantic similarity for summarisation purposes have either relied on a corpus-based approach or have implemented a new algorithm to compute the semantic similarity between words and short texts. Kenter & Rijke investigated whether it is possible to compute similarity between two short texts relying on just semantic features of the texts [11], using machine learning methods. Tsatsaronis et al, proposed a new approach for computing semantic relatedness between words based on a word thesaurus [23]. A framework presented in [12] proposes semantic role labeling for the summarization task. In a similar study, Aliguliyev [2] proposed a sentence-clustering approach for extractive document summarization, using an evolutionary algorithm and an appropriate ranking method.

[3]http://tldrlegal.com

Although examples like the above abound, the uniqueness of our target subject data precludes any method from being considered superior. EULAs are legal contracts and therefore have a more or less defined structure and terminology. Due to the nature of the target subject data, we have considered semantics-based clustering of extracts as a base for an intuitive Web user interface that supports human analysis of licenses. Rather than starting from scratch, we rely on an ontology-based Information Extraction (OBIE) method to extract specific excerpts from license documents. The relevant OBIE method [19] relies on the *Open Digital Rights Language* (ODRL) ontology[4], a mature and comprehensive vocabulary developed by a dedicated W3C Community Group. ODRL was designed specifically for digital content and its adequacy has been validated through a number of related efforts in the field [4, 8, 10, 22]. The existing OBIE method considers a number of classes from ODRL, namely the `Rule` class and its three subclasses: `Permissions`, `Prohibitions` and `Duties`. Properties of the `Rule` superclass include `action` (the operation relating to the asset) and `constraint` (constraints which affect the validity of actions). The OBIE method was implemented as a GATE OBIE pipeline, and is available for further adaption and extension. It can take as input legal text and generate three annotation types based on the above subclasses. The pipeline, which comprises common NLP tasks, an ontology-based gazetteer as well as a JAPE transducer for processing the text obtained an F-measure of over 70%, which is satisfactory since human inter-annotator agreement for the same task was calculated at 90%.

The above OBIE is in this paper taken as a basis for our approach. However, a number of shortcomings have been identified and have been addressed. In particular, the JAPE rules were extended to extract major features of policies for the clustering task. Moreover, since the ontology-based gazetteer in the pipeline annotates the concepts based on the root attribute of tokens, the morphological analyzer of the pipeline was improved to detect the stems more precisely. Next section provides more details regarding this issue.

Given that the OBIE results are satisfactory, we then consider an appropriate clustering method based on the input data: sets of `permissions`, `duties` and `prohibitions` together with a number of extracted features. In recent years, distributional semantic approaches have been extensively studied for word and text similarity [9, 13, 15]. Distributional semantic approaches benefit from the observation that words in the similar contexts tend to have similar meanings. A very popular distributional approach is word2vec and its extension paragraph2vec, which produces paragraph embeddings using deep learning technique [15]. However, since the authors have not released the source code and it was not possible to recreate their method due to incomplete descriptions, we could not implement their approach. Alternatively, DISCO is an open source Java application which retrieves the semantic similarity between short texts and phrases [13]. In addition, it allows users to build their own word embedding database from a text corpus. Over the years, DISCO has received a high community endorsement [3, 18, 20, 21]. Thus, our approach attempts to combine the extended OBIE-method with the DISCO clustering method for EULA summarisation.

[4]https://www.w3.org/ns/odrl/2/

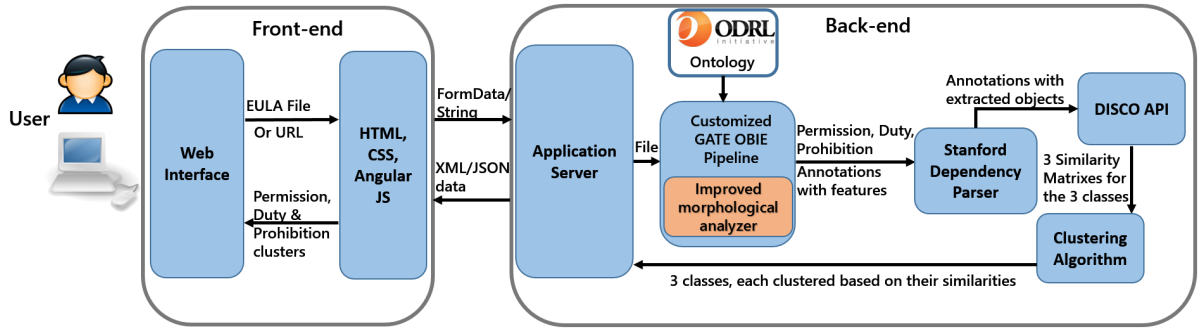

Figure 1: Architecture of the EULAide system

3 CLUSTERING BASED ON SEMANTIC SIMILARITY

In this section we explain our approach and its implementation in *EULAide*. Figure 1 shows the architecture of the framework. Each of the following subsections will explain how each contribution fits within this framework, i.e., i) modifications to the existing GATE OBIE pipeline, ii) word space creation for the proposed clustering method and iii) the *EULAide* service itself. The latter fits within the front-end, the former two within the back-end shown.

3.1 Modified OBIE pipeline for EULAs

Our efforts are based on modifications to the cited GATE OBIE pipeline tailored for processing EULAs. In an empirical study of 20 common licenses, we observed that many policy excerpts returned for each class could be thematically grouped into clusters. As an example, Table 2 shows three segments which have been extracted as `permissions` for the Apache License. The colored words have the same or very similar meaning and can therefore be grouped together. After clustering, users can focus on a summary for each cluster and if interested in that specific policy; browse all segments in each cluster.

Table 2: Example of Annotated Permissions in Apache

You may reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form.
You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work.

As mentioned in the previous section, the GATE OBIE pipeline component has been significantly improved. The improvements comprise:

i) extracting specific features from annotation types (e.g., actions, conditions and policy types),
ii) improving GATE morphological analyzer and
iii) adding the Stanford dependency parser [5] to the end of the pipeline for extracting the main object of each excerpt.

Table 3: Example of Features Extraction

If you join a Dropbox for Business account,	you must	use	it in compliance with your employer's terms and policies
condition		**action**	
each Contributor grants to You a	patent	license	to make, use, sell, import, and transfer (the Work)
	type of policy		**action**

The extracted features carry important information in EULAs and play an important role in clustering similar segments. Features include:

- the sequence of `action` for each segment, e.g., 'copy, reproduce', 'share', 'remove', etc.;
- the `condition` on which a specific `action` is granted or forbidden or obliged; and
- the `typeOfPolicy` which can be a 'copyright' or 'patent' or 'intellectual property right'.

Table 3 shows two examples of expected results of the feature extraction phase.

In order to extract `actions` more precisely, we added some rules in GATE morphological processing resource. This resource specifies the root of each token and in most cases, the stems of nouns are identified almost as the original noun itself, e.g., the lemmas for 'distributions', 'attribution' or 'attachment' are 'distribution', 'attribution' and 'attachment'. In this regard, the OBIE pipeline can not relate these words to the ontology concepts, because the ontology-based gazetteer annotates the text based on the root of each token. Consequently, the JAPE rules will fail to extract these tokens as `action` features. However, after customizing morphological analyzer, the accuracy of stem identification has improved significantly. The number of annotated concepts by the ontology-based gazetteer has increased from 9,630 to 9,927 for 20 licenses. As a result, the pipeline is now for example able to extract the following `actions` in: *"Activities other than **distribution** and/or **modification** of the Work are not covered by this license.."*.

In addition, the rational for adding dependency parser to the end of the pipeline was resolving the main object of each excerpt, in

order to generate a short and simple summary for each cluster by concatenating the 'action' and 'object' of all segments in one cluster. It is worth mentioning that integrating GATE with the Stanford NLP was quite a challenge, since both have defined quite different structures for annotations and related concepts in their APIs.

Once the three annotation type classes are built with the respective features, they are passed to a semantic similarity measurement component. This component builds a symmetric matrix for each class and passes it to the clustering algorithm. Finally, the clustering component groups the segments based on their similarities and the clustered `permissions`, `prohibitions` and `duties` are shown to the end-user. The next section provides more details regarding the similarity computation and its usage for text clustering.

3.2 Word Space Creation & Semantic Clustering

Out of a number of available open source tools supporting corpus-based semantic similarity between short texts, the DISCO API [13] coupled with its word space builder was identified as the most appropriate for our Java-based application. DISCO stands for extracting DIstributionally related words using CO-occurrences. Distributional approaches need a large corpus to build the word embeddings database. In order to create a domain specific word space for our approach, we used a dataset comprising 1.000 EULAs [14]. The dataset is passed to a method which generates a lemmatized text file consisting of three columns: token, a part-of-speech tag, and the base form (lemma). We execute the DISCO builder with the default configuration on the lemmatized file. The builder's output contains word vectors for each token. Finally, DISCO takes word space and two short texts as input and generates a real value between one and zero indicating the semantic similarity.

For computing the similarity between two short texts, DISCO uses the following formula:

$$Sim(T_1, T_2) = \frac{directedSim(T_1, T_2) + directedSim(T_2, T_1)}{2} \quad (1)$$

Assuming T_1 and T_2 consist of n word corresponding to w_{11}, ..., w_{n1} and w_{12}, ..., w_{n2}, the directed similarity is calculated as shown in Equation 2. The function $weight(word)$ returns a real value (between 0 and 1) for an input word. The weighting algorithm is based on frequency of the word in the corpus.

$$directedSim(T_1, T_2) = \sum_{i=1}^{n} weight(w_{i1}) * \max_{1 \le j \le n}[WordSim(w_{i1}, w_{j2})] \quad (2)$$

Algorithm 1 shows the sketch of our clustering approach. The semantic similarity is computed for all features of the extracted segments and the final similarity score is calculated by summing all four values. The rational behind the summation operation is being in line with the formula in Equation 2. Once, we obtained the similarity matrix for each class, a clustering algorithm is required to group similar segments. Agglomerative hierarchical clustering (HAC) is an established, well-known technique which has been shown to be a successful method for text and document clustering [1]. Furthermore, among different HAC methods, the average linkage has been proved to be the most suitable one for text categorization [1, 24]. Once the proper clustering technique is identified, we can pass similarity matrices to the clustering component. The HAC process continues until it reaches a pre-defined threshold.

Algorithm 1 Sketch of Semantic Clustering Algorithm

Require: permissions, prohibitions, duties with features
1: **for** the three classes **do**
2: **for** all segment pairs in each class **do**
3: $A \leftarrow$ similarity between actions ▷
4: $B \leftarrow$ similarity between conditions ▷
5: $C \leftarrow$ similarity between policy types ▷
6: $D \leftarrow$ similarity between the remainders of segments ▷
7: $finalSim \leftarrow A + B + C + D$ ▷
8: add finalSim to the corresponding matrix cell ▷
9: **end for**
10: do HAC clustering for the matrix with a threshold
11: **end for**
Ensure: clustered permissions, prohibitions & duties

3.3 *EULAide* Framework and Web Service

We developed a comprehensive license analysis framework *EULAide*, which also provides a comprehensive Web service interface[5]. As represented in the architecture, the client side Web interface implemented in Javascript using the AngularJS framework sends a request to the backend service and receives a JSON or XML object if the request is valid. In the back-end, several open-source APIs are orchestrated to provide an accurate result.

Figure 2 shows an example of *EULAide* output, applied to the Google terms of service[6]. The number of extracted excerpts by OBIE pipeline is 14, whereas the number of clusters has reduced to 9. The head of each accordion (block) contains the concatenation of 'action' and 'object' (extracted by Standford dependency parser) of all members in the cluster which can be replaced with a more proper summarization algorithm in future. As represented in the figure, the numbered excerpts are grouped together. In addition, there is a tooltip "more info" at the bottom of each block which the user can hover over and see the complete paragraph regarding that policy. The user can also see all the details by clicking on "Open All" button or choose to expand a specific accordion by clicking on the corresponding header (e.g., summary). Since *EULAide* is implemented using a two-layer architecture, it is platform independent, e.g., any client such as mobile apps can easily communicate to our server.

4 EXPERIMENTS

In order to evaluate the efficiency of *EULAide* as a summarization tool, three types of experiments were carried out. First we designed a test to measure the appropriateness of our clustering approach. Second we created an experiment to verify two hypotheses:

- *EULAide* needs less time and effort for EULA comprehension;
- semi-automatic information extraction and summarization may lead to information loss.

As the last experiment, the usefulness of *EULAide* perceived by users was estimated through a common usability test. The first

[5] http://butterbur17.iai.uni-bonn.de:8080/EULAideClient
[6] https://www.google.com/policies/terms/

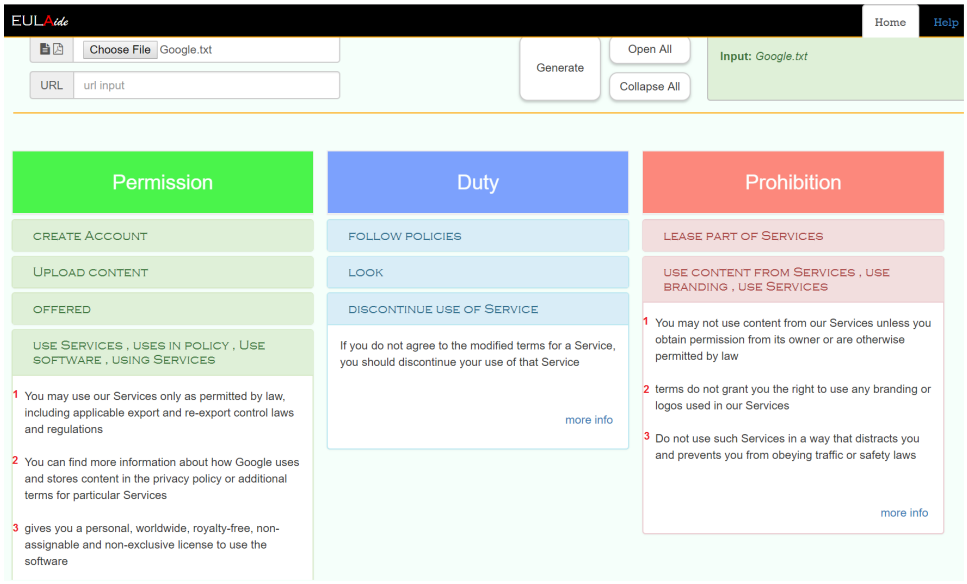

Figure 2: EULAide Platform Web interface showing the permission duty and prohibition clusters for a user provided EULA

experiment is presented in subsection 4.1 and the second and third experiments are reported jointly in subsection 4.2 .

4.1 The Clustering Approach Evaluation

4.1.1 Setup

The evaluation carried out investigates two hypotheses:

- to identify whether the proposed clustering method based on the OBIE-derived classes is generally useful (i.e. helps readers better digest and comprehend EULAs) and;
- to identify whether the devised feature-extraction method offers improved results;

When available, the quality of clustering methods is best measured by comparing machine-generated clusters with a reliable gold standard. However, considering the complexity and broadness of EULAs, it is very difficult to obtain or compile a suitable gold standard that is agreed and accepted by a majority. Therefore, as an alternative method we designed an experiment that can compare clustering preferences between human evaluators (to approximate inter-annotator agreement), and compare them with those generated computationally. To determine the usefulness of feature extraction, a second machine-generated clustering was included for a secondary comparison.

The input data for the evaluation consisted of a number of instances for the three EULA classes prohibitions, duties and permissions from four carefully-selected EULAs. The choice of the latter considered both a good balance between the three identified classes as well as sufficient brevity. EULAs that were very long were not suitable for this task since the human annotators' clustering task increases significantly in terms of complexity. The selected EULAs yielded the total number of class instances shown in Table 4. Here, one can note that the feature-based clustering method yields a marginally higher amount of clusters for two of the three classes

Table 4: Total Extracted Instances & Machine-generated Clusters with (Mf) and without (M) Considering Features

	#Instances	#Clusters-M	#Clusters-Mf
Permission	30	18	20
Duty	27	24	20
Prohibition	40	32	35

(e.g., `permission` and `prohibition`). A higher amount of clusters can be interpreted as a more fine-grained result. However, this can only be confirmed by comparison of these results with those of our human evaluators.

To carry out the human evaluation, five subjects were asked to cluster the OBIE-derived excerpts of `permission`, `prohibition` and `duty` within the context of each selected EULA. Since the target users of *EUALAide* are regular people who are not expected to be particularly acquainted with legal text and jargon, the five volunteers selected have different levels of higher education (under- and postgraduates) selected from a university campus. The only confirmed common interest between the individuals is an understanding of the need for EULA summarization methods such as the ones we propose. At the same time, to ensure that the task is properly and equally understood by all evaluators, an introduction to the *EULAide* tool, its vision, goals and the relevant concepts behind the input data was provided. However, the evaluators were not given instructions on how to cluster the results but were rather asked to devise their own clustering criteria as they best deemed fit. They were also explained that the input excerpts were semi-automatically extracted and could therefore contain some errors. The EULAs they considered contained and average of 7.5 `permissions`, 10 `prohibitions` and 6.7 `duties`.

Table 5: Clustering Results for Permissions

	h1	h2	h3	h4	h5	M	Mf
h1	(1)	0.9	0.65	0.83	0.9	0.86	0.8
h2	***	(1)	0.64	0.93	0.9	0.85	0.89
h3	***	***	(1)	0.65	0.65	0.7	0.72
h4	***	***	***	(1)	0.83	0.91	0.78
h5	***	***	***	***	(1)	0.85	0.88

Table 6: Clustering Results for Duties

	h1	h2	h3	h4	h5	M	Mf
h1	(1)	0.71	0.89	0.91	0.71	0.93	0.97
h2	***	(1)	0.61	0.63	0.95	0.7	0.68
h3	***	***	(1)	0.92	0.63	0.86	0.86
h4	***	***	***	(1)	0.65	0.85	0.88
h5	***	***	***	***	(1)	0.7	0.67

Table 7: Clustering Results for Prohibitions

	h1	h2	h3	h4	h5	M	Mf
h1	(1)	0.95	0.75	0.85	0.89	0.87	0.9
h2	***	(1)	0.75	0.85	0.92	0.91	0.92
h3	***	***	(1)	0.73	0.76	0.76	0.75
h4	***	***	***	(1)	0.89	0.8	0.81
h5	***	***	***	***	(1)	0.86	0.86

Table 8: Average Results

	Human	M	Mf
Permission	0.79	0.83	0.81
Duty	0.76	0.81	0.81
Prohibition	0.83	0.84	0.85

4.1.2 Human-Machine Comparisons & Discussion

The above experiment yielded two result sets: i) the two machine-generated clusters and ii) the 5 human-generated clusters. In order to compare and consider the two sets, we considered common methods for measuring clustering quality, i.e., the Rand index and F-measure [16]. Applying F-measure to clustering methods is similar to other information retrieval approaches, and considers (dis-)similarity of excerpts within clusters as a base for true/false positive/negatives. However, the F-measure disregards true negatives, or "the occurrence of dissimilar excerpts in different clusters". Thus, it does not take the proportion of correct non-clustering of unrelated excerpts into account; which is also very important in measuring the success criteria for this task. Alternatively, the Rand index formula shown in Equation 3 is used to measure the percentage of accuracy, which is more appropriate for our evaluation because it also factors true negatives.

$$RI = \frac{TP + TN}{TP + FP + FN + TN} \tag{3}$$

The interpretation of positive or negative decisions in clustering is drawn from decision series theory in arithmetic. Having n items, the space of all pairs of elements is computed by Equation 4.

$$\binom{n}{2} = \frac{n * (n-1)}{2} \tag{4}$$

In our results set we do not have one 'correct' clustering, but rather five subjective variations from each evaluator. Therefore, we applied the Rand index to calculate the cross-accuracy of clusters, using each evaluators' clusters in turns as the correct standard. We then simply considered the two machine-generated clusters as alternate clusters, and extended the cross-accuracy computations to cover both result sets. These computations generate a (symmetric) matrix of results, separated by class (`permissions`, `duties` and `prohibitions`), as shown in Table 5, Table 6 and Table 7.

The five human evaluators are enumerated h1-h5 and the two machine-generated clusters as M (without feature inclusion, i.e., passing the entire excerpts to the algorithm) and Mf (passing the entire excerpts with annotated features for clustering). The results shown per class indicate that there is a high-level of human agreement (ranging from a low of 0.61 to a high of 0.95). There is also a high-level of human-machine agreement (ranging from a low of 0.67 to a high of 0.97). This indicates that in general, there was higher disagreement between humans than between machine and humans. To further help with interpreting the above results, Table 8 enlists the agreement between the evaluators (Human), between the benchmark method and the evaluators (M) and between the feature-based clustering and the evaluators (Mf); once again organised per class.

The above results indicate that the applied clustering method based on the OBIE-derived instances is relatively successful, considering the level of human disagreement when performing the same task manually; but that there is no significant difference between the two clustering methods. On the other hand according to Table 8, the feature-based results are closer to human agreement. The accumulated deviation of **MF** from **Human** is 9%, whereas the aggregated deviation of **M** from **Human** is 10%. Although the difference is minor, the results reconfirms our previous assumption that feature-based approach generates more fined-grained clusters and is more attuned to human intuition and perception. While we are encouraged by the former result, we see the potential to further improve the feature-based clustering algorithm. The said features are already being used to improve the results shown in *EULAide* — the cluster titles shown in Figure 2 are based on the derived `actions`.

To improve the results of feature-based method, we will strive to more accurately extract the EULA-tailored feature. Since all of them are identified by semi-automatic rules, they are prone to some error. Increasing the accuracy of feature extraction phase will lead to better results.

4.2 *EULAide* Usability Experiments

4.2.1 Setup

In order to conduct our evaluation, the four EULAs from our previous experiment were again chosen for the current task. A legal expert was asked to design five multiple choice questions for each EULA (e.g., twenty in total). All questions are related to `permissions`,

Table 9: Average Time (In Seconds)

	Reading	Answering Phase1	Answering Phase2
EULA-Full	1185	75	152
EULA-*EULAide*	315	72	77

prohibitions and duties. Afterwards, six people from the university campus were selected to take part in our experiment. Each person studied all four EULAs: they had to read two EULAs in natural text and also exploits the *EULAide* service for the other two EULAs. With this setting, each EULA in natural text was read by three different students. Similarly, each EULA was browsed and studied with *EULAide* by three individuals. The participants were aware that their tasks are related to permissions, prohibitions and duties and they were asked to understand and digest EULA in order to answer the questions.

The question answering phase was split into two stages: first the participants had to answer the questions using their memory and without looking the EULA; Second, if they were unsure about some questions, they could check the EULA and use search tools to find the answer. The rational behind this setting was measuring how good users can remember policies and also how fast they can search for information in the EULA. The primary purpose of *EULAide* is to get an overview of an EULA before accepting it. In practice, when one is agreeing with terms and conditions, they should try to remember the important parts of license agreement in order to avoid infringing the regulations.

4.2.2 Evaluation & Discussion

In this section, we have reported the average results of our experiment, e.g., for each EULA - either in natural text mode or in *EULAide* mode - the average of three values (corresponding to three participants) was computed. In this case, we have eight average values: four EULAs in natural text mode plus the same four EULAs in *EULAide* mode. Finally, for simplicity and avoiding confusion, only the average value of each mode is presented here.

Table 9 shows the average time for each step of the experiment. According to the table, using *EULAide* to study and understand EULA takes significantly less time than reading the EULA in natural text, which was indeed an expected result. Furthermore, as already mentioned before, we have divided the answering phase into two steps: **phase1** is based on memory and **phase2** is allowing the users to search for the answer in the EULA. Not surprisingly, the average times of **phase1** are very similar, because regardless of the mode (natural text or *EULAide*), reading the questions and their multiple choice answers takes relatively equal time. However, regarding **phase2**, using *EULAide* to search for answers is one minute and 15 seconds faster than finding the desired information in the natural text license. Once again, this was an expected outcome, e.g., searching in a structured text is rather simpler.

The second part of current evaluation is concerned with the correctness of answers provided by the participants. Table 10 shows the average percentage of results. According to this table, the correctness of answers in the natural text mode is 5% higher than

Table 10: Average Percentage of Questions Results (%)

	Correct	Incorrect	Unanswered in Phase1		
			Phase2 Correct	Phase2 Incorrect	Phase2 Unanswered
EULA-Full	67	8	18.5	5	1.5
EULA-*EULAide*	62	15	6.5	4.5	12

EULAide mode, which is a reasonable result. If the end-user bears with the cumbersome legal lingua in the EULA and spends time studying it, he/she can understand the important parts of the license. However, as stated in the introduction, only a few people read EULA and *EULAide* is an attempt to motivate them to be aware of what they are agreeing to. Our result shows that if end-users exploit our tool, they can get on average 62% of the questions right, which is indeed very encouraging. Finally, there are some unanswered questions in **phase1** which leads us to the next phase. In the **phase2** of answering process, the participants were allowed to search for information in the EULA. According to the table, from 25% unanswered questions in the full text mode, people have found 18.5% correct answers after re-looking. On the other hand, from 23% unanswered questions in *EULAide* mode, only 6.5% of correct answers were found. Consequently, 12% of unsure questions remained unanswered. This is due to semi-automatic process of information extraction phase. The F-measure of OBIE pipeline is around 75% and not all of permissions, duties and prohibitions can be extracted with the pipeline. Therefore, not all of the questions were covered in *EULAide* and the participants could not find the answers.

In summary with *EULAide*, people get 19.5% incorrect answers, while with the full text and search, they get 13.3%, i.e., *EULAide* has on average 6% higher error rate (incorrect). Similarly, there are on average 10.5% more unanswered questions with our approach. While we expect that every (semi-)automatic approach leads to a potential information loss, we aim to verify that the error rate of *EULAide*, as well as its unanswered questions are consequences of automatic extraction and summarization. The information loss of (10.5+6)% by *EULAide* is a reasonable cost for the time gained (Exploiting *EULAide* is on average three times faster than the full text), and the increased incentive for familiarizing oneself with an EULA rather than simply accepting it. Furthermore, it should be stated here that the number of selected EULAs for this type of experiment was a bare minimum. EULAs are time-consuming and it is challenging to find people who are willing to read and understand them. We believe that the number of participants and selected EULAs are enough to indicative results, but also acknowledge that this experiment does not seek to make a scientific conclusion but rather an indication which is open to interpretation.

4.2.3 Usability Test

The last task of participants was filling a usability questionnaire for *EULAide* evaluation. We have reused a very common form available

Table 11: Average Scores of Six Participants for the Usability Questionnaire (Max=7)

Usefulness	Ease of Use	Ease of Learning	Satisfaction
6.14	6.11	6.75	6.0

online[7]. There are thirty questions categorized into four groups: 8 questions for usefulness, 11 for ease of use, 4 for ease of learning and 7 for satisfaction. There is seven options for each question ranging from 1 (strongly dissatisfied) to 7 (strongly satisfied). Table 11 shows the average scores of six participants for each category. The results are surely promising and implies that end-users are quite satisfied using *EULAide*. Furthermore, some participants stated a few points regarding the service. The positive feedbacks include: a nice and friendly user interface, fast response time (less than 1 minute), significant time reduction concerning EULA digestion, summary of each cluster, grouping similar policies and the ability to expand a specific cluster. The improvements given by participants suggest some interesting ideas for future work. Two participants recommended to include other aspect of EULA in the summary, e.g., what are the agreements between them and the service provider? Three users have said not all of `permissions`, `prohibitions` and `duties` are covered by *EULAide*. Last but not least, almost all participants were pleased with the summarization idea and encouraged us to improve the approach in future.

5 CONCLUSIONS & FUTURE WORK

We presented a holistic approach for the analysis and preparation of end-user license agreement. To the best of our knowledge, this is the first comprehensive approach for EULA interpretation and comprehension. The approach included an comprehensive ontology representing relevant terms, an ontology-based information extraction, a clustering method for excerpts and a Web-based user interface for self-service license analysis. Our evaluation showed that the clustering is effective and significantly reduces the number of relevant terms for users to focus on initially. In addition, according to the usability study response, *EULAide* is more visual and simpler to digest EULAs and saves around 75% of the time. However, we are aware that it comes at a marginal price of 10.5% loss of valuable information, which is an acceptable trade-off, considering the amount of time saved by users - especially since the full EULAs are not being read by a lot of users. We deem this work to be a significant step forward to make the description of rules and regulations governing online services, software tools, portals and apps more user friendly.

In future work, we aim to expand the application of the approach to other types of legal documents (e.g. contracts in general or regulatory documents). In addition, we aim to expand the work around the manually crafted ontology and automatically extracted terms and conditions into methodology for creation of comprehensive legal knowledge graphs. Furthermore, we aim to provide authors of EULAs means to accompany their licenses with a machine readable version, which would further increases the precision and recall of license analytics.

[7] http://garyperlman.com/quest/quest.cgi?form=USE

REFERENCES

[1] C. C. Aggarwal and C. Zhai. A survey of text clustering algorithms. In C. C. Aggarwal and C. Zhai, editors, *Mining Text Data*, pages 77–128. Springer, 2012.

[2] R. M. Aliguliyev. A new sentence similarity measure and sentence based extractive technique for automatic text summarization. *Expert Syst. Appl.*, 36(4):7764–7772, May 2009.

[3] S. Bhagwani, S. Satapathy, and H. Karnick. Semantic textual similarity using maximal weighted bipartite graph matching. In *Proceedings of the First Joint Conference on Lexical and Computational Semantics - Volume 1: Proceedings of the Main Conference and the Shared Task, and Volume 2: Proceedings of the Sixth International Workshop on Semantic Evaluation*, SemEval '12, pages 579–585, Stroudsburg, PA, USA, 2012. Association for Computational Linguistics.

[4] E. Cabrio, A. P. Aprosio, and S. Villata. These are your rights - a natural language processing approach to automated rdf licenses generation. In V. Presutti, C. d'Amato, F. Gandon, M. d'Aquin, S. Staab, and A. Tordai, editors, *ESWC*, volume 8465 of *Lecture Notes in Computer Science*, pages 255–269. Springer, 2014.

[5] D. Chen and C. D. Manning. A fast and accurate dependency parser using neural networks. In *EMNLP*, 2014.

[6] H. Cunningham, D. Maynard, and K. Bontcheva. *Text Processing with GATE (Version 8)*. Gateway Press CA, 2011.

[7] H. Cunningham, D. Maynard, and V. Tablan. JAPE: a Java Annotation Patterns Engine (Second Edition). Research Memorandum CS–00–10, Department of Computer Science, University of Sheffield, November 2000.

[8] E. Daga, M. d'Aquin, E. Motta, and A. Gangemi. *A Bottom-Up Approach for Licences Classification and Selection*, pages 257–267. Springer International Publishing, Cham, 2015.

[9] S. Deerwester, S. T. Dumais, G. W. Furnas, T. K. Landauer, and R. Harshman. Indexing by latent semantic analysis. *JOURNAL OF THE AMERICAN SOCIETY FOR INFORMATION SCIENCE*, 41(6):391–407, 1990.

[10] P. A. Jamkhedkar and G. L. Heileman. A formal conceptual model for rights. In *Proceedings of the 8th ACM Workshop on Digital Rights Management*, DRM '08, pages 29–38, New York, NY, USA, 2008. ACM.

[11] T. Kenter and M. de Rijke. Short text similarity with word embeddings. In *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*, CIKM '15, pages 1411–1420, New York, NY, USA, 2015. ACM.

[12] A. Khan, N. Salim, and Y. Jaya Kumar. A framework for multi-document abstractive summarization based on semantic role labelling. *Appl. Soft Comput.*, 30(C):737–747, May 2015.

[13] P. Kolb. Disco: A multilingual database of distributionally similar words. *Proceedings of KONVENS-2008, Berlin*, 2008.

[14] N. Lavesson, M. Boldt, P. Davidsson, and A. Jacobsson. Learning to detect spyware using end user license agreements. *Knowl. Inf. Syst.*, 26(2):285–307, Feb. 2011.

[15] Q. V. Le and T. Mikolov. Distributed representations of sentences and documents. In *Proceedings of the 31th International Conference on Machine Learning, ICML 2014, Beijing, China, 21-26 June 2014*, pages 1188–1196, 2014.

[16] C. D. Manning, P. Raghavan, and H. Schütze. *Introduction to Information Retrieval*. Cambridge University Press, New York, NY, USA, 2008.

[17] C. D. Manning, M. Surdeanu, J. Bauer, J. Finkel, S. J. Bethard, and D. McClosky. The Stanford CoreNLP natural language processing toolkit. In *Association for Computational Linguistics (ACL) System Demonstrations*, pages 55–60, 2014.

[18] N. K. Nagwani and S. Verma. A frequent term and semantic similarity based single document text summarization algorithm. *International Journal of Computer Applications (0975–8887) Volume*, pages 36–40, 2011.

[19] N. M. Nejad, S. Scerri, S. Auer, and E. M. Sibarani. Eulaide: Interpretation of end-user license agreements using ontology-based information extraction. In *Proceedings of the 12th International Conference on Semantic Systems*, SEMANTiCS 2016, pages 73–80, New York, NY, USA, 2016. ACM.

[20] C. Nguyen Ngoc, A. Roussanaly, and A. Boyer. Learning Resource Recommendation: An Orchestration of Content-Based Filtering, Word Semantic Similarity and Page Ranking. In *EC-TEL 2014 : 9th European Conference on Technology Enhanced Learning*, Open Learning and Teaching in Educational Communities, pages 302–316, Gratz, Austria, Sept. 2014. European Association of Technology Enhanced Learning, Springer.

[21] A. Qadir, P. N. Mendes, D. Gruhl, and N. Lewis. Semantic lexicon induction from twitter with pattern relatedness and flexible term length. In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence*, AAAI'15, pages 2432–2439. AAAI Press, 2015.

[22] S. Steyskal and A. Polleres. Defining expressive access policies for linked data using the odrl ontology 2.0. In *Proceedings of the 10th International Conference on Semantic Systems*, SEM '14, pages 20–23, New York, NY, USA, 2014. ACM.

[23] G. Tsatsaronis, I. Varlamis, and M. Vazirgiannis. Text relatedness based on a word thesaurus. *J. Artif. Int. Res.*, 37(1):1–40, Jan. 2010.

[24] Y. Zhao, G. Karypis, and U. Fayyad. Hierarchical clustering algorithms for document datasets. *Data Mining and Knowledge Discovery*, 10(2):141–168, 2005.