EULAide: Interpretation of End-User License Agreements using Ontology-Based Information Extraction

Najmeh Mousavi Nejad
Institute for Informatics
University of Bonn, Germany
nejad@cs.uni-bonn.de

Simon Scerri
Fraunhofer IAIS & Institute for Informatics
University of Bonn, Germany
scerri@iai.uni-bonn.de

Sören Auer
Fraunhofer IAIS & Institute for Informatics
University of Bonn, Germany
auer@cs.uni-bonn.de

Elisa M. Sibarani
Institute for Informatics
University of Bonn, Germany
sibarani@iai.uni-bonn.de

ABSTRACT

Ignoring End-User License Agreements (EULAs) for online services due to their length and complexity is a risk undertaken by the majority of online and mobile service users. This paper presents an Ontology-Based Information Extraction (OBIE) method for EULA term and phrase extraction to facilitate a better understanding by humans. An ontology capturing important terms and relationships has been developed and used to guide the OBIE process. Through a feedback cycle we have improved its domain-specific coverage by identifying additional concepts. In the detection and extraction, we focus on three key rights and conditions: `permission`, `prohibition` and `duty`. We present the EULAide system, which comprises a custom information extraction pipeline and a number of custom extraction rules tailored for EULA processing. To evaluate our approach, we created and manually annotated a corpus of 20 well-known licenses. For the gold standard we achieved an Inter-Annotator Agreement (IAA) of 90%, resulting in 193 `permissions`, 185 `prohibitions` and 168 `duties`. An evaluation of the OBIE pipeline against this gold standard resulted in an F-measure of 70-74% which, in the context of the IAA, proves the feasibility of the approach.

Keywords

End-User License Agreements; EULA; Ontology-Based Information Extraction; Inter-Annotator Agreement; IAA

1. INTRODUCTION

The ubiquitous availability of the Internet results in a massively growing number of online and mobile services for end-users, ranging from personal information management (e.g., Web mail, calendar, address book), cloud storage (e.g., photo/video repositories) over collaboration tools (e.g., document authoring, messaging) to e-commerce (online shops, song/movie subscription services). Everyday new services emerge and their providers aim at quickly increasing the user base and market share by providing a user-friendly interfaces to the services; frequently even permitting users to use the service completely free of charge. In most cases, users have to accept terms and conditions governing the usage before utilizing such services. However, their use remains regulated through detailed terms and conditions and not infrequently users are unaware of their obligation to 'pay' for the service by sharing their personal data and contributions.

According to a *Fairer Finance survey*[1], the 'small print' provided by some companies now runs to over 30,000 words (the length of a short novel) and unsurprisingly, fewer than a third of those asked said they read the terms and conditions. In order to partially relieve users, information extraction and text analytics techniques can be applied to recognize, classify and present certain kinds of text in End-User License Agreements (EULAs). We introduce a novel technique that exploits knowledge encoded in an ontology for annotating, extracting and classifying EULA content into predefined categories, leveraging Ontology-based Information Extraction (OBIE). OBIE guides Information Extraction (IE) pipelines to process unstructured or semi-structured natural language text by exploiting ontologies to extract pre-defined structured information and annotating the text using ontology terms. OBIE was favoured for this approach for a number of reasons. Primarily, the reliance on a vocabulary engineered by domain experts grounds our work in existing standards and broadens its application. For example, our work can benefit initiatives undertaken by the *Permissions & Obligations Expression Working Group*[2]. Secondly, as a form of legal document license agreements tend to have clear structures and terminologies, sometimes even containing identical clauses and phrases. This facilitates the 'mapping' of natural-language text to machine-readable conceptualisations in the ontology for our use-case.

Following a survey of existing vocabularies we identified

SEMANTiCS 2016, September 12-15, 2016, Leipzig, Germany
 ISBN 978-1-4503-4752-5/16/09…$15.00
DOI: http://dx.doi.org/10.1145/2993318.2993324

[1]Survey available at:
`http://www.bbc.com/news/business-27109000`
[2]`https://www.w3.org/2016/poe/charter`

the *Open Digital Rights Language* (ODRL) ontology[3], developed by the W3C ODRL Community Group as the most appropriate basis, based on its maturity and comprehensiveness. Despite being created specifically for digital content, the ODRL is broad enough to be used for different types of resources (such as linked open data, digital works, online services, etc.). Furthermore, the adequacy of ODRL has been validated through a number of related efforts in the field [3, 14, 11, 7]. In this phase of our project we have focused on the `Rule` class defined by ODRL, since it is an "abstract common ancestor to `Permissions`, `Prohibitions` and `Duties`". Some properties of `Rule` include `action` (the operation relating to the asset) and `constraint` (constraints which affect the validity of actions). Since the three relevant subclasses of `Rule` (`permission, prohibition, duty`) inherit these properties, the ontology satisfies our needs. However, through a feedback cycle we have improved its domain-specific coverage by identifying additional instances (around 50).

Similarly, out of a number of available Natural Language Processing (NLP) tools supporting OBIE methods, the GATE framework coupled with its ANNIE IE system [5] and its support for customized Java Annotation Pattern Engine (JAPE) grammar rules [6] was identified as the most appropriate for the following reasons. First, OBIE is supported even at the level of hand-coded grammar rules (via JAPE). Second, it still offers one of the easiest methods for embedding a GATE pipeline in Java. Third, it has strong evaluation tools for NLP including inter-annotator agreement and F-measure calculation based on a gold standard, thus relieving the developer from combining different software solutions for one single task.

Based on our customized ODRL extension and the GATE/ ANNIE software framework, we present in this article the EULAide system, which comprises a custom processing pipeline and a number of extraction rules tailored for EULA processing. Our contributions are in particular:

- A comprehensive ontology based on the ODRL ontology capturing knowledge related to permissions, prohibitions and duties.
- A software architecture and implementation for EULA information extraction comprising linguistic pre-processing, inclusion of an ontology-based gazetteer and a large number of custom extraction rules.
- A detailed evaluation using an assessment framework including 20 of the most popular EULAs, which were manually annotated and demonstrate the feasibility of the approach with an overall F-measure of 0.7 - 0.74.

To the best of our knowledge, in spite of the importance of the issue there exists no other automated framework or program offering similar IE methods for EULAs. As is explained in section 2, although there have been numerous efforts addressing EULAs, the objectives were somewhat different. In particular, we demonstrate the value of using an ontology for this task. In section 3 the OBIE pipeline is presented in detail. The evaluation results are presented in section 4. We conclude the article in section 5 by also outlining plans to extend EULAide to cover more complex policies in addition to the basic rights and conditions targeted in this paper.

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

2. RELATED WORK

We compare and contrast our approach with efforts in two main categories: (1) EULA information extraction methods, including some that are driven by ontologies, and (2) other relevant application-independent OBIE techniques.

Some researchers have targeted EULA Information Extraction for the benefit of end-users. An example is the `tldrlegal`[4] online service, which uses a manual, crowdsourced way to help people understand the most commonly-used licenses. It is supported by users and everyone can create an account and suggest a short summary of a chosen license and then upload the EULA to the website. Our approach strives to bypass the need for substantial human input and generate similar results for any input EULA.

Some existing efforts have applied vocabularies for EULA annotation, with various degrees of success. Before we compare them with EULAide, we list the surveyed vocabularies in Table 1. As can be observed, the domain coverage ranges from very specific (MPEG-21, e-commerce applications) to very broad (digital rights, open data, linked data). ODRL stands out not only in terms of being the most current (most recently updated vocabulary), but also in terms of comprehensiveness (ranking 2nd from the domain-independent vocabularies). ODRL has also demonstrated the highest community endorsement. Examples included the RDF licenses database[5], which is a first attempt at developing a knowledge base of licenses, and which combines the Creative Commons Rights Expression Language (CC REL) and ODRL to express EULAs as RDF. Furthermore, supplementary efforts have identified ODRL 2.0 as the best candidate for defining policies in linked data licenses [14]. In [11] a formal conceptual model based on CC REL, XrML and ODRL was integrated into a platform. The license picker ontology also exploits ODRL [7].

The Semantic Copyright project and the copyright ontology respectively, provide rich representation for the attributes of digital work and even support basic reasoning over the uses allowed, limited by only capturing basic rights [1]. Unfortunately, there is little documentation and no additional information about this effort and the ontology. SemanticLIFE is a desktop framework that allows users to make a call to a semantic repository while installing new software (either open source or proprietary) [2]. The repository can be managed by an administrator and contains license agreement ontologies in OWL format. Users download the corresponding license ontology on his/her semantic desktop and it is matched with user-based policies that are also captured in an ontology. Although the idea is interesting, the paper does not present and report any use-case or evaluation of the approach.

The NLL2RDF Framework exploits NLP techniques to generate RDF expressions of license agreements [3], targeting open linked data as their primary use-case. The authors use ODRL and CC REL vocabulary to manually annotate the dataset and build a gold standard. Similarly, NLL2RDF also is primarily concerned with `Permissions (derive, reproduce, modify, copy, sell), Prohibitions (commercialise) and Duties (shareAlike, attachPolicy, attribute)`. However, in contrast to our OBIE method, NLL2RDF employs a supervised machine-learning algorithm

[4] `http://tldrlegal.com`

[5] `http://datahub.io/dataset/rdflicense`

Table 1: Vocabularies and Ontologies for EULAs

Name	Domain Coverage	# Classes & Instances	# Properties	Last Release
CC REL[6]	linked data	28	42	2013/11
ODRL[7]	open digital content	85	56	2015/03
LDR[8] (derived from ODRL)	linked data resources	77	21	2014/09
LiMO[9]	open data	12	60	2013/05
L4LOD[10]	web of data	16	5	2013/05
ODRS[11]	open data	2	15	2013/07
MPEG-21 Rights Data Dictionary[12]	contains the terms as standardized in ISO/IEC 21000-6	2000 standardized terms having the characteristics of a structured ontology		2005/07
Copyright Ontology[13]	digital rights management	99	42	2014/01
IPROnto[14]	intellectual property rights with a focus on e-commerce applications	113	54	2003/12
Semantic Copyright - Basic[15]	works in digital format	67	32	2009/10
Semantic Copyright - Registry[16]	works in digital format	126	48	2009/10

to classify EULAs.

The framework's limitation is that it only covers a limited number of rights and conditions. Furthermore, notwithstanding that their dataset covered 37 licenses, the class with the highest frequency only scored 28 occurrences. This low number might be related to their training data, since after going through their publicly available dataset[17] we noticed a scarce number of annotations in the 4-5 page licenses.

OBIE has been investigated or applied to a wide-number of use-cases [16, 15, 8, 13]. In this section, we focus on the four most comprehensive recent studies.

A review of OBIE applications was provided in [16]. The authors define key factors that characterise OBIE systems and provide a comparison of OBIE applications and their different specifications, including evaluation metrics and methods. The authors identify the most widely-used tools for OBIE to be GATE, sProUT[18] and the Stanford CoreNLP[19]. In a more recent survey [13] ontology learning and population is explored and several steps towards completing this task are listed. Different approaches for OBIE are also introduced and explained, including rule-based methods, machine learning algorithms, parse trees and hybrid systems.

Taking cue from observations that combining multiple IE methods can increase the performance of an IE pipeline [15], a hybrid system combining extraction rules and machine learning-based IE methods was developed. This system is based on an architecture for *Ontology-Based Components for Information Extraction* (OBCIE) [8], whose ambition is to enable researchers to adapt OBIE by benefiting from its modularity characteristic. Using a heuristic algorithm, an ontology-based error detection mechanism that can recognize sentences that are inconsistent with the ontology was also implemented. The combination of OBIE with more conventional IE methods is also an avenue we plan to explore in the near future.

3. ONTOLOGY-BASED INFORMATION EXTRACTION FOR EULAS

In this section, we describe the architecture and implementation of the EULAide system, which builds on the GATE framework. GATE provides three types of resources: *Language Resources* (LRs) which collectively refers to data; *Processing Resources* (PRs) which are used to refer to algorithms; and *Visualization Resources* (VRs) which represent visualization and editing components. In future, we will also take advantage of GATE's support for Java-embedded pipelines by simply including GATE libraries JAR files.

3.1 Architecture

Figure 1 illustrates the architecture, whose core is the EULAide GATE Pipeline specifically tailored for EULA processing. The inputs for this pipeline is an EULA in natural language text, and the EULAide ontology based on an ODRL ontology extension. The pipeline consists of (1) a linguistic pre-processing stage, (2) an ontology-based Gazetteer, and finally (3) the main OBIE transducer. The *Linguistic Pre-processor* consists of the following PRs:

- *Document Reset:* Clears all the annotations that were assigned before.
- *Tokeniser:* Adds two annotation sets: *Token* and *space-Token*.

[6] http://creativecommons.org/ns
[7] https://www.w3.org/community/odrl/
[8] http://oeg-dev.dia.fi.upm.es/licensius/static/ldr/
[9] http://data.opendataday.it/LiMo/
[10] http://ns.inria.fr/l4lod/
[11] http://schema.theodi.org/odrs/
[12] http://iso21000-6.net/view/rddDictionary.php
[13] http://rhizomik.net/html/ontologies/copyrightonto/
[14] http://dmag.ac.upc.edu/ontologies/ipronto/
[15] http://www.semanticcopyright.org/index.php/ontology/basic
[16] http://www.semanticcopyright.org/index.php/ontology/copyright-registry
[17] http://www.airpedia.org/nll2rdf/dataset-licenses/
[18] http://sprout.dfki.de
[19] http://stanfordnlp.github.io/CoreNLP/

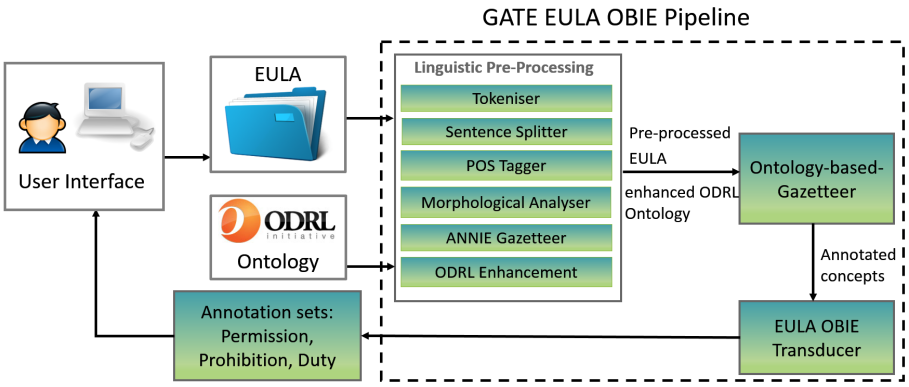

Figure 1: High-level architecture of the EULAide system

- *Sentence Splitter:* Splits the sentences and creates an annotation set *Sentence*.
- *POS Tagger:* Applies Part-of-Speech tagging and adds a feature called *category* to each *Token* annotation.
- *Morphological Analyser*: Inserts a new feature to each *Token*, called *root*. Later the *ontology-based Gazetteer* annotates the concepts based on the *root* feature.
- *ANNIE Gazetteer:* Reuses existing relevant lists (e.g., countries) but adds additional lists covering terms that carry important information in license agreements, such as file formats and different synonyms for the term 'license' and 'asset'.
- *ODRL Enhancement:* This semi-automatic phase takes the standard ODRL as input and updates a local representation with additional instances.

The last component in the processor merits further justification. Following a number of trials and iterations, we have extended the ODRL instance base (focusing on the `Rule` subclasses (`Permission, Prohibition and Duty`)) with a number of instances that were routinely observed in our corpus but were missing from the original vocabulary definitions. Although the examples observed are semantically-comparable with the ODRL definitions, they are not specifically covered as instances. As a result, due to the failure of the Gazetteer to detect them, the required patterns could not be matched. For example, `delete` is an instance of the ODRL `Action` class, and is described as: *"The Assigner requires that the Assignees permanently removes all copies of the Asset"*. This semantically matches the use of the `destroy` or `remove` keywords when used in the same context, i.e., users should destroy or completely remove the software from their devices. Although these two keywords were observed on numerous occasions, the standard ODRL-driven gazetteer could not detect them. In total, more than 50 such keywords were identified, including common EULA terms such as `attach`, `submit`, `send`, `import`, `add`, `destroy`, `remove`, etc. The *ODRL Enhancement* PR is required to include these instances in the run-time representation of ODRL. It takes the form of a JAPE Transducer, which runs customised JAPE grammars in a semi-automatic process. JAPE Grammars are composed of rules that match linguistic patterns on the left-hand side (LHS) to the annotations to be created on the right-hand side (RHS). Technically, the *ODRL Enhancement* transducer matches ontology-related and keywords on the LHS to Java codes for adding instances to the ontology in the RHS.

The next major pipeline component is the *Flexible Ontology-based Gazetteer*, which takes the enhanced ODRL as input and creates a new annotation set called *Lookup* containing matches to ODRL instances and concepts. Basically, this PR matches any text features to the (extended) ODRL element labels, flexibly allowing for various inflections. The result is a list of semantic annotations pairing bits of text to the matching ODRL elements.

The final and most important PR is the customised *EULA OBIE Transducer*, which considers all the previous annotation sets as inputs and matches pre-defined annotation patterns to the final annotation sets: (`Permission, Prohibition, Duty`). This PR is described in detail in the next section.

3.2 EULA OBIE Transducer

The transducer executes in 10 phases and builds on all outputs from the previous stages to create the annotations. We have implemented 15 grammar rules to generate the final `Permission`, `Prohibition` and `Duty` annotation sets. The definition of the JAPE rules is heavily-based on ODRL community specification documentations[20] where each class and subclass is explained in detail. For instance, according to the vocabulary documentation, `include` is an instance of `Action` class and means: *"The Assigner requires that the Assignee(s) include(s) other related assets in the Asset."*. Therefore, the presence of '`include`' in a sentence can suggest the presence of a `Duty` in an EULA.

In the sequel, we explain the main phases in more detail:

annotateClasses.

This phase separates the *Lookup* annotation set which contains all the ontology-derived annotations. Since, in this project phase, we deal with basic rights, the focus is on a specific part of the ontology. According to ODRL community group explanations [10], we have differentiated two main categories for actions: *DutyAction* and *Permission-ProhibitAction*. Although some actions are present in both annotations (like `delete`), this separation is a vital step for the next phases. Apart from actions, important words which

[20] `https://www.w3.org/community/odrl/vocab/2.1/`

Table 2: Example of a Permission as extracted by EULAide

ANNIE Gazetteer	**This license** License	grants you	to copy, share and reproduce	**the product** Asset
Ontology based Gazetteer	This license	grants you	**to copy, share and reproduce** Lookups	the product
Annotate Classes Phase	This license	**grants** you Perm Words	**to copy, share and reproduce** Perm Actions	the product
Extract Permissions	**This license** License	**grants you** Perm Words	**to copy, share and reproduce** Obj (Perm Actions)+	**the product** Asset

carry significant information for rights detection were also determined. For instance, *must* and *should* are labeled with *DutyWords*; *may*, *can*, *grant*, *permit*, etc. are labeled with *PermissionWords* and similarly *may not*, *can not*, *not allowed*, *prohibited*, etc. are labeled with *ProhibitionWords*.

extractPermissions.

Since there may be different structures of sentences in a license, we implemented 4 rules for the extraction of permissions. For instance the sentence *"[You] [may] [copy, share and reproduce] [the product]"* will fire the following grammar rule: ('+' means one or more occurrences)

`[Subj][permWords][permAction]+[Asset]`

On the other hand, the sentence *"[This license] [grants] [you] [to copy, share and reproduce] [the product]"* will fire another rule:

`[License][permWords][Object][permAction]+[Asset]`

It should be clarified that some annotation sets such as *License* and *Asset* are detected by our own-defined *ANNIE gazetteers*. Table 2 shows the different steps towards extracting of the above `permission`. After the pre-processing phase, first the *ANNIE gazetteer* generates two annotation sets: *License* and *Asset*. Second, the *ontology-based gazetteer* annotates the concepts based on the ontology with *Lookup* label. Then the first phase of *EULA OBIE transducer* is executed and the *PermWords* and *PermAction* annotation sets are created. Finally the *Permission1* rule from the second phase fires and annotates the whole sentence as a `permission`.

extractProhibitions.

This phase is very similar to the previous one, except that in the grammar rules the *PermissionWords* are replaced with *ProhibitionWords*. Therefore the sentence *"[You] [may not] [copy, share and reproduce] [the product]"* will be annotated as a `prohibition`.

extractDuties.

For extracting the duties there are more diverse structures, hence we implemented five different rules, one of which is the following:

`[Subj][DutyWords][DutyAction][obj_clause][Asset]`

This rule fires when processing the sentence:*"[You] [must] [include] [a copy of this License] [with your product]"*

clean.

In this phase, all intermediate annotation sets are deleted from the output and only the three final annotation sets are retained: `Permission`, `Prohibition`, `Duty`.

4. EXPERIMENTS

Currently our method does not require any training set and the information extraction relies solely on the described JAPE rules and gazetteer entries derived from the ontology. However, a solid test set was required in order to

- manually extract relevant grammatical and lexical patterns and translate them into JAPE rules;
- use the test set as a base for an inter-annotator agreement test to identify realistic upper-bounds of successful automatic extraction;
- evaluate the OBIE approach by comparing the pipeline execution F-score to the level of inter-annotator agreement achieved.

In the next sections we explain how we set about to achieve a gold standard following an inter-annotator agreement experiment, and discuss the evaluation set-up and results of the OBIE approach.

4.1 Gold Standard Creation

Although some EULA datasets are available, neither of them is annotated at the required level of granularity. For example, some RDF expressions of EULAs are freely available[21]. However, they were not useful for our experiments for two reasons. Primarily, a comprehensive dataset should contain two kinds of licenses: license templates (e.g., the core definition of the GNU Public License) and actual instances (e.g., an Apple Inc. EULA). Unfortunately, the available dataset includes only the core definitions. Secondly, there are no structural links between the annotations and the text and therefore we could not use it for our evaluation purposes. The only annotated corpus that was suitable for our objectives from a requirement point-of-view was the NLL2RDF project dataset[22]. However, despite containing 37 annotated licenses the frequencies of annotations are too low and the annotations were deemed incomplete if not unreliable. For instance, no `Prohibition` or `Duty` annotations, and only four `Permission` annotations were observed for the Mozilla license. In contrast, after a careful manual annotation we identified and verified (following consultation with experts) 16 `Duties`, 12 `Permissions` and 3 `Prohibitions`.

In order to compile our own gold standard, we collected 20 popular EULAs in their original text. Table 3 shows the details of the input set, including the average word and character count. When choosing the licenses, we intentionally tried to cover varying ones both in terms of structure and content. For instance, since Mozilla and Netscape are offered by the same organization, their EULAs carry a lot of identical phrases. Furthermore, we selected licenses of varying lengths, purposely avoiding ones that are too short. The average word count of a license within the corpus is 3,206 and the average character count without space is 16,815.

To prepare the gold standard, two annotators familiar with EULA-like text annotated the corpus independently following an introduction to the relevant ODRL concepts. Using the dedicated GATE plugin the Inter-Annotator Agreement (IAA) score was then computed for the two annotation

[21] `http://datahub.io/dataset/rdflicense`

[22] `http://www.airpedia.org/nll2rdf/dataset-licenses/`

Table 3: Specification of Licenses

Row	Name	Word Count	Character Count (excl. whitespace)
1	Apache	1,581	8,652
2	Apple Website	3,328	19,831
3	bitTorrent	4,383	23,215
4	Dropbox	1,938	9,859
5	Eclipse	1,701	9,534
6	EUP License	2,087	10,709
7	Facebook	4,404	22,435
8	GNU	5,614	28,386
9	Google	1,869	9,501
10	iTunes Boss	965	5,272
11	Jetbrains	1,833	10,230
12	LaTeX	3,011	15,377
13	Minecraft	1,962	8,489
14	Mozilla	2,821	14,829
15	Netflix	5,134	27,097
16	Python	1,440	8,034
17	Red hat	3,303	17,085
18	Skype	4,365	22,571
19	SoundCloud	8,927	46,779
20	Sun	3,450	18,406

sets. The plugin offers two types of IAA measurement: F-measure and agreement based on the kappa statistic. The latter has been criticized and has a number of well-known limitations. Kappa is suitable when annotators have the same number of instances but with different class labels. It is not recommended for text mark-up tasks, such as named entity recognition and information extraction [9]. When the annotators themselves determine which text spans they can annotate, the F-measure should be used. The F-measure has been less controversial and is also indicated as the most appropriate IAA measure in the GATE manual itself, given the nature of our annotation task [5].

GATE offers three ways to measure IAA: the *Strict* measure only takes the annotations that have exactly the same span in the text and considers all partially correct annotations as incorrect; the *Lenient* measure considers all partially correct annotations as correct; and the *Average* measure allocates a half weight to partially correct annotations (i.e. it takes the average of strict and lenient). Since our goal is to extract blocks of text representing pre-defined concepts, we do not required fine-grained results. Therefore, the lenient measure was deemed sufficient since we simply want to guide human readers to which parts of the text contain permissions, duties and prohibitions.

An initial lenient IAA resulted in an acceptable F-score of 77%. Considering the complexity of EULA texts, the initial IAA is well within the reasonable performance range described in related literature for IE tasks of a similar complexity [4]. The annotators were then invited to discuss their disagreements with the aim of conflict resolution. The results of this discussion also contributed to additional improvements to our customised gazetteer and JAPE grammars. After this consultation phase the IAA increased to a very satisfactory 90%. Table 4 shows the lenient IAA results. In order to produce a final gold standard for evaluating the actual EULAide pipeline, we removed the 10% disagreements and only retained the agreed-on annotations. The final gold standard contains 193 `permissions`, 185 `prohibitions` and 168 `duties`.

Table 4: Lenient IAA for Two Annotators

	Precision	Recall	F-measure
Permission	0.94	0.90	0.92
Prohibition	0.79	0.94	0.86
Duty	0.86	0.96	0.91
Summary	0.87	0.93	0.9

4.2 Evaluation & Discussion

In this section, two types of experiments are presented: one without the ontology enhancement phase and the other including this phase. We have utilized the Corpus Quality Assurance tool in GATE. The tool calculates precision, recall and F-score between two annotation sets in a corpus. Similar to IAA, lenient F-measure was selected for the comparison of EULAide with the gold standard. Furthermore, among different types of F-scores, we decided to rely on F2-score, since in the legal documents domain and EULA as a specific type of legal texts, it is risky not to detect some `prohibitions` and `duties`. For instance, if there is a license agreement in a hospital, it is crucial to identify all types of `prohibitions` and `duties` and missing important policies may cause adverse effects. However, in order to provide a complete overview of EULAide performance, all three types of F-scores are reported here.

Table 5 and Table 6 show the evaluation results. As represented in the tables, the ontology enhancement transducer has significantly improved the results. Although this phase has increased the recall by 19%, it has also decreased the precision by 4%. This means that while adding more concepts to the ontology is helpful for the IE process, the concepts should be inserted to the ontology carefully and with contemplation.

According to the Table 6, the precision for `permission` is 74%. For example, the following sentence is wrongly annotated by EULAide: *"this License grants you permission to propagate or modify any covered work"*. At the first glance, the annotation seems to be correct but actually the complete paragraph is: *"However, nothing other than this License grants you permission to propagate or modify any covered work. These actions infringe copyright if you do not accept this License. Therefore, by modifying or propagating a covered work, you indicate your acceptance of this License to do so."*. Although this paragraph seems to contain a `permission`, the main phrase granting the `permission` has appeared in the previous paragraphs and is the following sentence: *"You may make, run and propagate covered works"*. According to the annotators, this sentence which EULAide wrongly recognized as a `permission`, defines a new type of policy known as an `Agreement`. Therefore, for future work we will enrich EULAide with different types of policies and consequently achieving higher accuracy.

EULAide has missed some `permission` annotations and therefore the recall value is 75%. While the definition of our JAPE rules is based on the ODRL ontology, some of the current `permissions` in EULAs do not follow the ontology definition. For example, this sentence was annotated as a `permission` in the gold standard: *"The number of permitted*

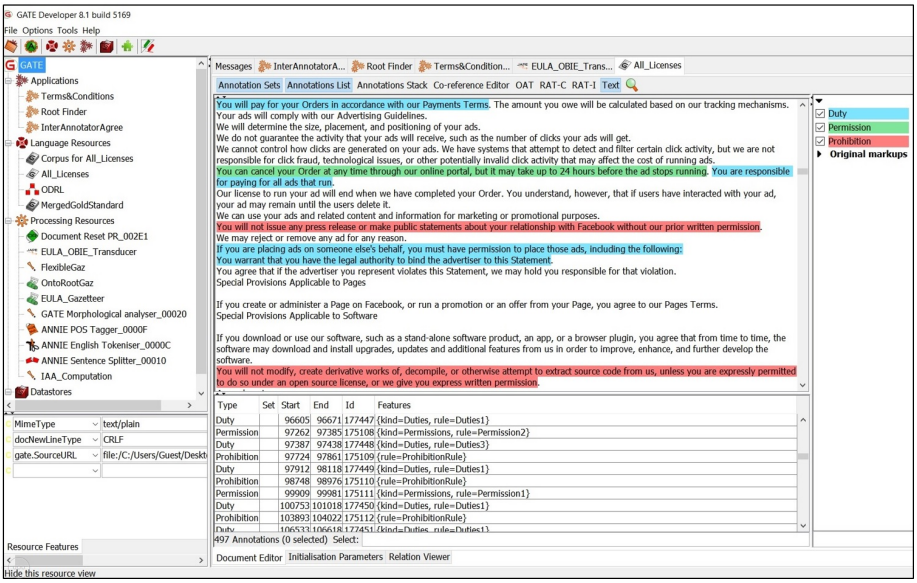

Figure 2: The output of EULAide indicating duties (blue), permissions (green) and prohibitions (red).

Table 5: Evaluation of EULAide without Ontology Enhancement

	Precision	Recall	F0.5	F1	F2
Permission	0.75	0.56	0.71	0.64	0.59
Prohibition	0.89	0.47	0.75	0.61	0.52
Duty	0.73	0.43	0.64	0.54	0.46
Overall	0.79	0.49	0.7	0.6	0.53

Table 6: Evaluation of EULAide with Ontology Enhancement

	Precision	Recall	F0.5	F1	F2
Permission	0.74	0.75	0.74	0.74	0.75
Prohibition	0.89	0.63	0.82	0.74	0.66
Duty	0.66	0.67	0.67	0.67	0.67
Overall	0.75	0.68	0.74	0.72	0.7

participants on a group video call varies from 3 to a maximum of 10, subject to system requirements.". In this case it is difficult to identify an `action` that would also satisfy the definition of `action` in ODRL (e.g., *"An action is the operation relating to the asset for which permission is being granted"*). The problem arises from the fact that ontologies can not always cover all needs. Most ontologies are useful because they provide the basic modeling, but they need to be extended for some use-cases. Although ODRL is one of the best ontologies for EULA domain, it does not cover 100% of the needs for EULA classification. In our future work, we will compile our own ontology based on ODRL as the foundational model and will add new classes and concepts to increase the coverage area of the ontology.

For the `prohibition` annotations, we achieved a precision of 89% and a recall of 63%. Some of the incorrect `prohibitions` annotated by EULAide can be traced back to the annotators disagreements in the IAA experiment. As an example, one of the disagreements in the IAA was the following sentence: *"Do not use such Services in a way that distracts you and prevents you from obeying traffic or safety laws."*. While one of the annotators believes that this is a `soft prohibition` for the user, the other considers this phrase as a warning to the licensee and believes that this sentence does not prohibit the users from accessing the service. However since it was not a both-agreed annotation, it was removed from the gold standard and therefore is counted as one of the incorrect annotations generated by EULAide. This problem can be solved by including a new annotation set called `soft prohibition`. In our future work, we can annotate all the warnings with this new label.

To increase the recall for the `prohibition`, we should consider the language variability in EULAs. In implementing the rules we have realized most `prohibitions` include the terms *"You must/should/may not"*, but unsurprisingly some of them have more complex structures (e.g., *"No one other than Sun has the right to modify the terms applicable to Covered Code created under this License"*). Therefore, including different possible structures of a sentence in a rule would lead to a higher recall.

Finally, the precision for `duty` annotation is 66%. While we had assumed that the terms *"it is your responsibility"* or *"you are responsible"* along with some other patterns will lead to a `duty` in an EULA, the annotators did not mark all of them as `duties`. For instance, EULAide annotated the sentence *"Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program"* as a `duty`. On the contrary, the annotators believe that this is more like a warning to the user and does not mean an obligation the user should do in order to receive a specific `permission`. However, they both agree that the sentence *"You are responsible for ensuring that the Source Code version remains available"* is a `duty`.

The recall value for `duty` is 67% and shows that EULAide did not detect 33% of the duties. Similar to the `prohibition`, several structures of a natural language makes the IE process complex. For example, in most EULAs, the licensor usually uses the words *"You must/should/have to"* for defining a `duty`, but on the other hand there are some

with different forms stating an obligation (e.g., *"The GNU GPL requires that modified versions be marked as changed, so that their problems will not be attributed erroneously to authors of previous versions"*). In future work, we will define new annotation set to catch the name of the EULA which is passed as an input to the pipeline. Having such information about the title of the license, we can implement new rules detecting the above sentence as a `duty`.

Figure 2 shows an example of EULAide output, applied to *Facebook* terms of use. In total, EULAide has identified 196 `permissions`, 131 `prohibitions` and 170 `duties`. We are now working on creating a platform-independent application in Java (leveraging GATE JAR files) to generate a summary of an EULA using extractive summarization methods based on our annotation sets.

5. CONCLUSIONS & FUTURE WORK

In this article, we presented an approach and its implementation for automatic ontology-based annotation of licenses and End-User-License-Agreements. In an era of a proliferation of online services, facilitating easy and transparent user knowledge of the terms and conditions as laid out in the license agreements can not be overestimated. The domain of licenses and EULAs is well suited for automated information extraction, since the legal language used is more constrained and standardized than arbitrary text content. Also, the ontology-based background knowledge enables us to achieve a relatively high precision and coverage with an overall F-measure of more than 70%. We show that improvements to the ontology directly and significantly affect the annotation and extraction results.

Regarding future work, first we intend to improve EULAide accuracy further based on the findings in the evaluation. We also plan to extract more policies and rights (e.g., `conditions, agreements, constraints`, etc.) from EULAs. Furthermore, since we have already started improving ODRL, we are confident with further enhancement, a new and more comprehensive ontology can be published based on ODRL ontology. Combining different IE techniques and applying a synonym finder of vocabularies are other future directions for system improvement. Last but not least, we will implement our pipeline in Java and will design a web/mobile application where a user can copy the EULA text into it and the application will generate a short summary which is also a machine-readable version based on the ontology.

Acknowledgments

This work is supported in part by the European Union under the Horizon 2020 Framework Program for the projects BigDataEurope (GA 644564) and European Data Science Academy (GA 643937).

6. REFERENCES

[1] Semantic copyright project, http://semanticcopyright.org.

[2] M. Ahmed, A. Anjomshoaa, M. A. e yar, A. M. Tjoa, and A. Khan. Towards an ontology-based solution for managing license agreement using semantic desktop. In *ARES*, pages 309–314. IEEE Computer Society, 2010.

[3] E. Cabrio, A. Palmero Aprosio, and S. Villata. *The Semantic Web: Trends and Challenges: 11th International Conference, ESWC 2014, Anissaras, Crete, Greece, May 25-29, 2014. Proceedings*, chapter These Are Your Rights, pages 255–269. Springer International Publishing, Cham, 2014.

[4] H. Cunningham. Information extraction, automatic. In *Encyclopedia of Language and Linguistics*, pages 665–677. 2005.

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

[6] 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.

[7] E. Daga, M. d'Aquin, E. Motta, and A. Gangemi. *The Semantic Web: ESWC 2015 Satellite Events: ESWC 2015 Satellite Events, Portorož, Slovenia, May 31 – June 4, 2015, Revised Selected Papers*, chapter A Bottom-Up Approach for Licences Classification and Selection, pages 257–267. Springer International Publishing, Cham, 2015.

[8] F. Gutierrez, D. Dou, S. Fickas, D. Wimalasuriya, and H. Zong. A hybrid ontology-based information extraction system. *Journal of Information Science*, page 0165551515610989, 2015.

[9] G. Hripcsak and A. S. Rothschild. Agreement, the f-measure, and reliability in information retrieval. *Journal of the American Medical Informatics Association*, 12(3):296–298, 2005.

[10] R. Iannella. Open digital rights language (odrl) version 1.1. *W3c Note*, 2002.

[11] 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.

[12] 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.

[13] R. Shah and S. Jain. Ontology-based information extraction: An overview and a study of different approaches. *International journal of computer Applications*, 87(4), 2014.

[14] 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.

[15] D. C. Wimalasuriya and D. Dou. Components for information extraction: Ontology-based information extractors and generic platforms. In *Proceedings of the 19th ACM International Conference on Information and Knowledge Management*, CIKM '10, pages 9–18, New York, NY, USA, 2010. ACM.

[16] D. C. Wimalasuriya and D. Dou. Ontology-based information extraction: An introduction and a survey of current approaches. *J. Inf. Sci.*, 36(3):306–323, June 2010.