Legal text processing within the MIREL project

Milagro Teruel*, **Cristian Cardellino***, **Fernando Cardellino***, **Laura Alonso Alemany***, **Serena Villata**†
*Universidad Nacional de Córdoba Argentina
INRIA & Université Côte d'Azur CNRS France

Abstract

We present the roadmap and advances in the area of Information Extraction from legal texts within the EU-funded MIREL project (MIning and REasoning with Legal texts). We describe the resources and tools we have developed for Natural Language Processing in the legal domain, i.e., annotated corpora and automated classifiers for Named Entity Recognition and Linking and Argument Mining. Our final objective is to identify arguments, their content and the relations between them in legal text, with a proof-of-concept in judgments of the European Court of Human Rights (ECHR), to finally support reasoning tasks over mined argumentative structures. This representation will arguably be useful for applications like a reading aid, enhanced information retrieval, structured summarization, intelligent search engines or information extraction. All tools and resources are available at `https://github.com/PLN-FaMAF/legal-ontology-population` and `https://github.com/PLN-FaMAF/ArgumentMiningECHR`.

Keywords: Argument Mining, Named Entity Recognition, Classification and Linking, Legal Information Extraction

1. Introduction and Motivation

Automated legal text processing is becoming more and more relevant within legal practice. According to the MIT Technology Review, the U.S. Consultancy group McKinsey estimates that 22% of a lawyer's job and 35% of a law clerk's job can be automated (Winick, 2017), for example:

> "JPMorgan announced earlier this year that it is using software called Contract Intelligence, or COIN, which can in seconds perform document review tasks that took legal aides 360,000 hours." "CaseMine, a legal technology company based in India, builds on document discovery software with what it calls its "virtual associate," CaseIQ. The system takes an uploaded brief and suggests changes to make it more authoritative, while providing additional documents that can strengthen a lawyer's arguments."
>
> (Winick, 2017)

Natural Language Processing (NLP) tools have the capabilities scan huge amounts of legal documents, identify portions relevant to a given case and even present them in an orderly manner for a lawyer needs to craft a case, more quickly and more exhaustively than humans given the huge amount of data to process. In case law, if law practitioners are provided with relevant cases when they are building their arguments for a new case, they could be more liable to produce a sounder argumentation. It is also to be expected that cases are resolved more definitely if compelling jurisprudence is provided, even at an early stage in the judicial process. More and more technological solutions are being developed in this line, which shows the feasibility and utility of this line of work.

One of the objectives of the MIREL project[1] is to develop tools for MIning and REasoning with Legal texts, with the aim of translating these legal texts into formal representations that can be used for querying norms, compliance checking, and decision support. Open-source tools and resources are also very important to provide equality in the access to the law. However, developing such tools is costly. Tools are usually trained with examples that have been manually analyzed and annotated by a domain expert, so we aim to reduce the cost of developing such tools by taking advantage of existing annotated resources.

In this paper, we present our roadmap and advances to develop such tools, working in two main areas: Named Entity Recognition, Classification and Linking (NERC and NEL) and Argument Mining (Lippi and Torroni, 2016). For each of these two areas, we both built annotated datasets following precise guidelines, and experimented supervised and unsupervised learning methods. More precisely, we have built a tool for NERC and NEL in the legal domain by exploiting the Wikipedia as an annotated corpus. To retrieve the relevant portion of the Wikipedia, we have established a mapping between an ontology of the legal domain, LKIF (Hoekstra et al., 2007), and an ontology covering the Wikipedia knowledge, YAGO (Suchanek et al., 2007). We have also explored the use of different flavors of word embeddings to transfer a Wikipedia-based model to judgments of the ECHR. We present extensive evaluation of the tools. For Argument Mining, we are manually annotating a corpus of judgments of the ECHR, with the focus on inter-annotator agreement and the performance of automatic analyzers to approach a balance between the descriptive adequacy and the performance of analyzers.

In the following Section, we outline the roadmap of our proposal, and then we go on to describe the tools and resources we are developing for NERC and NEL (Section 3.) and for Argument Mining (Section 4.), comparing them with the existing approaches in these domains. Conclusions end the paper.

2. Objectives of Information Extraction within MIREL

The final goal within the Information Extraction area of MIREL is to obtain a representation of legal texts that shows their arguments and anchors them semantically. To do that, our main subgoals are:

[1] `http://mirelproject.eu/`

- identify Named Entities and link them to a domain ontology, thus providing semantics, and
- identify argument components and their relations.

Argument Mining aims to discover the argumentative structure of a text. In the case of judgments, understanding the argumentative structure is crucial for legal actors (attorneys, judges) to make a judgment actionable in other legal actions, for example, to use the judgment as case-law. However, Argument Mining is a difficult task, even more so in the legal domain, where texts have very complex syntactical structures and semantic distinctions are very precise. Moreover, Argument Mining does not specifically deal with the propositional content of argument components. Identifying arguments does not usually include obtaining a subject-matter representation of the content of components (vs. their discursive, argumentative representation). However, for targeted applications, for higher-level analysis and for reasoning techniques we require that the propositional content is integrated with argumentative information. To do that, we build upon Information Extraction techniques.

Information Extraction is typically implemented in a pipeline. The first building block of this pipeline is usually Named Entity Recognition and Classification (NERC). The extension of NERC that anchors Named Entities to external knowledge bases, like ontologies, is known as Named Entity Linking (NEL). There are many domains where NERC achieves a good performance, and it has been shown to have a very positive impact in many applications (information retrieval, machine translation), even without any other Information Extraction technique. In particular for the legal domain, it has been shown to positively impact the identification of claims in legal texts (Surdeanu et al., 2010).

We consider the relation between NERC and Argument Mining within legal texts analogous to that of NERC and event detection in non-argumentative texts, like biomedical articles. Indeed, in both cases NERC provides an anchor to ontology-based semantics, but the relation between higher-level units is left for some other module. In factual texts, the relevant unit is the fact, which can be more or less equaled to a proposition. In contrast, in argumentative texts the relevant unit is the argument component, which can be thought of as the basic building block for applications like a reading aid, information retrieval, structured summarization.

NERC and NEL are highly domain-dependent tasks. That is why a legal NERC/NEL requires specific resources. However, developing such resources specifically for the legal domain is very costly. We have implemented a low-cost approach to legal NERC and NEL that takes advantage of the Wikipedia as an annotated corpus, more concretely, of the portion of the Wikipedia that belongs to the legal domain. To do that, we have implemented a mapping between an ontology of the legal domain, LKIF, and the YAGO ontology that is linked to the Wikipedia. This has resulted in the additional benefit of populating LKIF, which is a rather abstract ontology, and enriching its connection to Linked Open Data at more levels than the top of the ontology.

The workflow of our approach to analyze arguments in legal texts is as follows:

1. pre-process documents
2. identify and classify Named Entities
3. anchor Named Entities to a domain ontology
4. syntactico-semantic analysis of sentences, propositional representation
5. identify argument components
6. identify relations between argument components

The result of this process will be a useful input for applications like reading aids, information retrieval, structured summarization or reasoning.

In what follows we describe the tools and resources we are developing to deal with NERC and NEL and Argument Mining in the legal domain.

3. Named Entity Recognition and Linking

In this section we describe our approach to NERC and NEL.

In the legal domain, Named Entities are not only names of people, places or organizations, as in general-purpose NERC. Named Entities are also names of laws, of typified procedures and even of concepts. Named Entities may also be classified differently, for example, countries and organizations are classified as *Legal Person*, as can be seen in the following example extracted from a judgment of the European Court of Human Rights[2]:

Example 3..1 *The* $[Court]_{organization}$ *is not convinced by the reasoning of the* $[$*combined divisions of the Court of Cassation*$]_{organization}$*, because it was not indicated in the* $[judgment]_{abstraction}$ *that* $[$*Eğitim-Sen*$]_{person}$ *had carried out* $[$*illegal activities*$]_{abstraction}$ *capable of undermining the unity of the* $[$*Republic of Turkey*$]_{person}$*.*

We take an unexpensive approach to build a NERC/NEL system, by exploiting the information already available in Wikipedia as annotated examples, and connecting it with an ontology of the legal domain. More concretely, we aligned the WordNet- and Wikipedia-based YAGO ontology[3] (Suchanek et al., 2007) and the LKIF ontology[4] (Hoekstra et al., 2007) specifically conceived for representing legal knowledge. By doing this, we are transferring the semantics of LKIF to Wikipedia entities and populating the LKIF ontology with Wikipedia entities and their mentions. At the same time, we obtain a high number of manually annotated examples, taking linked strings in the Wikipedia as examples of entity mentions. With these examples, we can automatically learn a Named Entity Recognizer, Classifier and Linker.

We see that, while results on Wikipedia documents are good, there is a drop in performance when we change the domain and apply NERC to judgments of the European Court of Human Rights (ECHR). To deal with this domain change, we have explored the usage of word embeddings, without much improvement. After an analysis of error, we have identified a number of factors that will most probably impact in significant improvements.

[2] Extracted from the case Eğitim ve Bilim Emekçileri Sendikası v. Turkey, ECHR, Second Section, 25 September 2012, http://hudoc.echr.coe.int/eng.

[3] www.yago-knowledge.org/

[4] http://www.estrellaproject.org/lkif-core/

3.1. Aligning ontologies to acquire examples from the Wikipedia

The Wikipedia is a source of manually annotated examples, if we consider linked strings in the Wikipedia as examples of entity mentions. To gain access to those examples in the Wikipedia that belong to the legal domain, we aligned the WordNet- and Wikipedia-based YAGO ontology[5] (Suchanek et al., 2007) and the LKIF ontology[6] (Hoekstra et al., 2007) of the legal domain.

On the one hand, LKIF (Hoekstra et al., 2007) is an abstract ontology describing a core of basic legal concepts, with a total of 69 law-specific classes. It covers many areas of the law, but it is not populated with concrete real-world entities. On the other hand, YAGO is a knowledge base automatically extracted from Wikipedia, WordNet, and GeoNames, and linked to the DBpedia ontology[7] and to the SUMO ontology[8]. It represents knowledge of more than 10 million entities, and contains more than 120 million facts about these entities, tagged with their confidence. This information was manually evaluated to be above 95% accurate.

In our alignment process, we do not map relations but only classes. The mapping was carried out using the following methodology: for each LKIF concept, we try to find an equivalent in YAGO. If there is no direct equivalent, then we try to find a subclass, if not, a superclass. When some equivalent concept has been found, we establish the alignment using the OWL primitives `equivalentClass` and `subClassOf`. Finally, we navigate YAGO to visit the related concepts and check whether they could be aligned with another LKIF concept or if they were correctly represented as children of the selected concept. This implies that some legal concepts in YAGO are not in our ontology because they were not represented in LKIF. This is the case, for example, of the subdomain of *Procedural Law* or *Crime*, which were two annotate entities in the judgments of the ECHR. We can expect that whenever the ontology is applied to a specific subdomain of the law, it will need to be extended with the relevant concepts.

Of 69 law-specific classes within the LKIF ontology, 30 could be mapped to a YAGO node, either as children or as equivalent classes, thus 55% of the classes of LKIF could not be mapped to a YAGO node, because they were too abstract (i.e., *Normatively_Qualified*), there was no corresponding YAGO node circumscribed to the legal domain (i.e., *Mandate*), there was no specific YAGO node (i.e., *Mandatory_Precedent*), or the YAGO concept was overlapping but not roughly equivalent (as for "*agreement*" or "*liability*").

From YAGO, 47 classes were mapped to a LKIF class, with a total of 358 classes considering their children, and summing up 4'5 million mentions. However, the number of mentions per class is highly skewed, with only half of YAGO classes having any mention whatsoever in Wikipedia text.

[5] `www.yago-knowledge.org/`

[6] `http://www.estrellaproject.org/lkif-core/`

[7] `http://wiki.dbpedia.org/`

[8] `http://www.adampease.org/OP/`

The LKIF and YAGO ontologies are very different, and the task of NERC and NEL also differ from each other. In order to assess the performance of the classification at different levels, we established some orthogonal divisions in our ontology, organized hierarchically and effectively establishing different levels of granularity for the NERC and NEL algorithms to work with.

1. NER (2 classes): The coarsest distinction, it distinguishes NEs from non-NEs.
2. NERC (6 classes): Instances are classified as: Abstraction, Act, Document, Organization, Person or Non-Entity.
3. LKIF (69 classes, of which 21 have mentions in the Wikipedia): Instances are classified as belonging to an LKIF node.
4. YAGO (358 classes, of which 122 have mentions in the Wikipedia): Instances are classified as belonging to the most concrete YAGO node possible (except an URI), which can be either child of a LKIF node or an equivalent (but it is never a parent of an LKIF node).
5. URI (174,913 entities): Entity linking is the most fine-grained distinction, and it is taken care of by a different classifier, described in Section 3.3..

Example 3..1 can be tagged for NEL as follows:

Example 3..2 *The* $[\textit{Court}]_{European\_Court\_of\_Human\_Rights}$ *is not convinced by the reasoning of the* $[\textit{combined divisions of the Court of Cassation}]_{YargitayHukukGenelKurulu}$*, because it was not indicated in the* $[\textit{judgment}]_{Court\_of\_Cassation's\_judgment\_of\_22\_May\_2005}$ *that* $[\textit{Eğitim-Sen}]_{Education\_and\_Science\_Workers\_Union\_(Turkey)}$ *had carried out* $[\textit{illegal activities}]_{\emptyset}$ *capable of undermining the unity of the* $[\textit{Republic of Turkey}]_{Turkey}$.

The mapping between LKIF and YAGO is available at `https://github.com/PLN-FaMAF/legal-ontology-population`.

To build our corpus, we downloaded a XML dump of the English Wikipedia[9] from March 2016, and we processed it via the WikiExtractor (of Pisa, 2015) to remove all the XML tags and Wikipedia markdown tags, but leaving the links. We extracted all those articles that contained a link to an entity of YAGO that belongs to our mapped ontology. We considered as tagged entities the spans of text that are an anchor for a hyperlink whose URI is one of the mapped entities. We obtained a total of 4,5 million mentions, corresponding to 102,000 unique entities. Then, we extracted sentences that contained a mention of a named entity.

3.2. Learning a NERC

Using this corpus, we trained a classifier for Named Entity Recognition and Classification. The objective of this classifier is to identify in naturally occurring text mentions the Named Entities belonging to the classes of the ontology,

[9] `https://dumps.wikimedia.org/`

and classify them in the corresponding class, at different levels of granularity.
We have applied different approaches to exploit our annotated examples: a Support Vector Machine (SVM), the Stanford CRF Classifier for NERC (Stanford NLP Group, 2016), and a neural network with a single hidden layer, smaller than the input layer. We have explored more complex configurations of the neural network, including Curriculum Learning (Bengio et al., 2009), a learning strategy that is specially adequate for hierarchically structured problems like ours, with subsequent levels of granularity. However, none of these more complex configurations improved performance. For more details about the use of Curriculum Learning in our NERC, refer to (Cardellino et al., 2017).

3.3. Learning a NEL

The Named Entity Linking task consists in assigning YAGO URIs to the Wikipedia mentions. The total number of entities found in the selected documents is too big (174,913) to train a classifier directly. To overcome this problem, we use a two-step classification pipeline. Using the NERC provided by the previous step, we first classify each mention as its most specific class in our ontology. For each of these classes, we train a classifier to identify the correct YAGO URI for the instance using only the URIs belonging to the given class. Therefore, we build several classifiers, each of them trained with a reduced number of labels. Each classifier is trained using only entity mentions for a total of 48,353 classes, excluding the 'O' class.
The classifiers learnt for each of the classes were Neural Network classifiers with a single hidden layer, of size 2*number of classes with a minimum of 10 and a maximum of 500. Other classifiers, in particular, the Stanford NERC, cannot handle the high number of classes.
As a comparison ground, we also evaluated two baselines, a random classifier and a k-nearest neighbors. For the random baseline, given the LKIF class for the entity (either ground truth or assigned by an automated NERC), the final label is chosen randomly among the YAGO URIs seen for that LKIF class in the training set, weighted by their frequency. The k-nearest neighbors classifier is trained using the current, previous and following word tokens, which is equivalent to checking the overlap of the terms in the entity.
We distinguish two types of evaluations: the performance of each classifier, using ground truth ontology classes, and the performance of the complete pipeline, accumulating error from automated NERC. The individual classifier performance is not related to the other classifiers, and is affected only by the YAGO URIs in the same LKIF class. It is calculated using the test set associated with each class, that does not include the 'O' class.

3.4. Word Embeddings for Transfer Learning

The experiments were also carried out using word embeddings. Word embeddings provide a representation of words that counters the overfitting that is found in small corpora. Word embeddings are known to be particularly apt for domain transfer, because they provide some smoothing over the obtained model, preventing overfitting to the training set. Therefore, we expect them to be useful to transfer the models obtained from Wikipedia to other corpora, like the judgments of the ECHR.
However, it is also known that embeddings are more adequate the bigger the corpus they are learnt from, and if the corpus belongs to the same domain to which it will be applied. In our case, we have a very big corpus, namely Wikipedia, that does not belong to the domain to which we want to apply the embeddings, namely the judgments. Therefore, we have experimented with three kinds of embeddings: embeddings obtained from Wikipedia alone (as described above), those obtained with the same methodology but from the judgments alone, and those obtained with a mixed corpus made of judgments of the ECHR, and a similar quantity of text from Wikipedia.
The Wikipedia embeddings were obtained from the corpus we later use for the NERC task. To train word embeddings for judgments of the ECHR, we obtained all cases in English from the ECHR's official site available on November 2016, leading to a total of 10,735 documents.
All embeddings were trained using Word2Vec's skip-gram algorithm. All words with less than 5 occurrences were filtered out, leaving roughly 2.5 million unique tokens (meaning that a capitalized word is treated differently than an all lower case word), from a corpus of 1 billion raw words. The trained embeddings were of size 200, and taking them we generate a matrix where each instance is represented by the vector of the instance word surrounded by a symmetric window of 3 words at each size. Thus, the input vector of the network is of dimension 1400 as it holds the vectors of a 7 word window total.

3.5. Performance of NERC and NEL

To evaluate the performance, we computed accuracy, precision and recall in a word-to-word basis in the test portion of our Wikipedia corpus, totalling 2 million words of which the half belong to NEs and the other half to non-NEs.
For this particular problem, accuracy does not throw much light upon the performance of the classifier because the performance for the majority class, non-NE, eclipses the performance for the rest. To have a better insight on the performance, the metrics of precision and recall are more adequate. We calculated those metrics per class, and we provide a simple average without the non-NE class. Besides not being obscured by the huge non-NE class, this average is not weighted by the population of the class (thus an equivalent of macro-average). Therefore, the differences in these metrics are then showing differences in all classes, with less populated classes in equal footage with more populated ones.
Evaluating on Wikipedia has the advantage that NERC and NEL models have been learnt with Wikipedia itself, so they are working on comparable corpora. However, even if it is useful to detect NEs in the Wikipedia itself, it is far more useful for the community to detect NEs in legal corpora like norms or case-law. That is why we have manually annotated a corpus of judgments of the European Court of Human Rights, identifying NEs that belong to classes in our ontology or to comparable classes that might be added to the ontology. This annotated corpus is useful to evaluate the performance of the developed NERC and NEL tools,

but it will also be used to train specific NERC and NEL models that might be combined with Wikipedia ones.
More precisely, we annotated excerpts from 5 judgments of the ECHR, obtained from the Court website[10] and totalling 19,000 words. We identified 1,500 entities, totalling 3,650 words. Annotators followed specific guidelines, inspired in the LDC guidelines for annotation of NEs (Linguistic Data Consortium, 2014). Annotators were instructed to classify NEs at YAGO and URI levels. The annotated documents are available at `https://github.com/PLN-FaMAF/legal-ontology-population`.

3.5.1. NERC results on Wikipedia

approach	accuracy	precision	recall	F1
NER (2 classes)				
SVM	1.00	.54	.06	.11
Stanford NER	.88	.87	.87	.87
NN	1.00	**1.00**	**1.00**	1.00
NN+WE	.95	.95	.95	.95
NERC (6 classes)				
SVM	.97	.37	.18	.24
Stanford NER	.88	.78	.82	.79
NN	.99	**.89**	**.83**	**.86**
NN+WE	.94	.84	.78	.81
LKIF (21 classes)				
SVM	.93	.53	.26	.35
Stanford NER	.97	**.84**	**.71**	**.77**
NN	.97	.73	.65	.69
NN+WE	.93	.67	.60	.63
YAGO (122 classes)				
SVM	.89	.51	.25	.34
Stanford NER	–	–	–	–
NN	.95	**.76**	**.64**	**.69**
NN+WE	.90	.68	.61	.64

Table 1: Results for Named Entity Recognition and Classification on the test portion of the Wikipedia corpus.

The results for NERC on the test portion of our Wikipedia corpus at different levels of abstraction are reported in Table 1. We show the overall accuracy (taking into consideration the 'O' class), and the average recall, precision and F-measure across classes other than the non-NE class. The Stanford NERC could not deal with the number of classes in the YAGO level, so it was not evaluated in that level. We also show results with handcrafted features and with word embeddings obtained from the Wikipedia.
At bird's eye view, it can be seen that the SVM classifier performs far worse than the rest, and also that word embeddings consistently worsen the performance of the Neural Network classifier. The Stanford NERC performs worse than the Neural Network classifier at the NER level, but they perform indistinguishably at NERC level and Stanford performs better at LKIF level. However, it can be observed that the Neural Network performs better at the YAGO level than at the LKIF level, even though there are 122 classes at the YAGO level vs. 21 classes at LKIF level.

[10] `hudoc.echr.coe.int`

3.6. NERC results on the judgments of the ECHR

The results for NERC in the corpus of judgments of the ECHR are shown in Table 2. We can see the results with the models trained on Wikipedia and applied to the ECHR documents, and with models trained with and applied to the ECHR corpus (divided in training and test splits). We can also see models working on different representations of examples. The variations are handcrafted features and different combinations of embeddings: obtained from Wikipedia alone, obtained from the judgments of the ECHR alone, and obtained from Wikipedia and the ECHR in equal parts.
We can see that, on the ECHR corpus, results obtained for models trained with the annotated corpus of ECHR judgments perform significantly better than those trained with Wikipedia, even if the latter are obtained with a much bigger corpus. This drop in performance is mainly due to the fact that the variability of entities and the way they are mentioned is far smaller in the ECHR than in Wikipedia. There are fewer unique entities and some of them are repeated very often (e.g., "Court", "applicant") or in very predictable ways (e.g., cites of cases as jurisprudence).
For models trained with the annotated corpus of ECHR judgments, word embeddings decrease performance. This results are mainly explainable because of overfitting: word embeddings prevent overfitting, and are beneficial specially in the cases of very variable data or domain change, which is not the case when the NERC is trained with the ECHR corpus, with very little variability.
We also highlight that there is little difference between word embeddings trained with different inputs, although Wikipedia-trained word embeddings present better performance in general. There is no consistent difference between mixed and ECHR trained embeddings. In contrast, in Wikipedia-trained models, ECHR and mixed (ECHR+Wikipedia) word embeddings improve both precision and recall. This shows that, when we have a domain-specific model, embeddings obtained from a significantly bigger corpus are more beneficial. However, when no in-domain information is available, a representation obtained from many unlabeled examples yields a bigger improvement. For a lengthier discussion of these results, see Teruel and Cardellino (2017) (Teruel and Cardellino, 2017).

3.7. NEL results on Wikipedia

NEL could not be evaluated on the corpus of judgments, but only on Wikipedia, because annotation at the level of entities has not been consolidated in the corpus of judgments of the ECHR. Therefore, approaches to NEL have only been evaluated on the test portion of the corpus of Wikipedia.
Results are shown in Table 3. As could be expected from the results for NERC, word embeddings worsened the performance of prediction. We can see that the performance of NEL is quite acceptable if it is applied on ground-truth labels, but it only reaches a 16% F-measure if applied over automatic NERC at the YAGO level of classification. Thus, the fully automated pipeline for NEL is far from satisfactory. Nevertheless, we expect that improvements in YAGO-level classification will have a big impact on NEL.
We also plan to substitute the word-based representation of

		NERC (6 classes)				LKIF (21 classes)				YAGO (122 classes)			
		Acc.	Prec.	Rec.	F1	Acc.	Prec.	Rec.	F1	Acc.	Prec.	Rec.	F1
Wiki trained	NN	**.76**	**.56**	**.24**	**.25**	**.76**	**.13**	**.07**	**.08**	**.76**	**.06**	**.03**	**.03**
	NN+WE wiki	.73	.34	.21	.21	.74	.08	.05	.05	.74	.03	.02	.02
	NN+WE mix	.75	.42	.23	.23	.75	.10	.06	.06	.75	.04	.04	.03
	NN+WE echr	.75	.38	.24	.24	.75	.11	.07	.07	.74	.04	.03	.03
	Stanford	.73	.36	.17	.16	.73	.07	.06	.05	-	-	-	-
ECHR trained	NN	**.80**	**.69**	.41	.47	**.81**	.46	.24	.28	**.81**	**.33**	.18	**.21**
	NN+WE echr	.77	.52	.54	.52	.75	.27	.32	.27	.79	.22	.22	.19
	NN+WE wiki	.78	.54	**.58**	.55	.79	.30	**.34**	.29	.80	.24	.22	.19
	NN+WE mix	.77	.48	.50	.48	.77	.28	.32	.28	.78	.23	**.22**	.18
	Stanford	.79	.67	.51	**.56**	**.81**	**.49**	.30	**.34**	.80	.28	.21	**.21**
	K-NN	.73	.54	.49	.50	.73	.32	.27	.25	.72	.22	.18	.16

Table 2: Results for Named Entity Recognition and Classification on the corpus of judgments of the ECHRwith models trained only with the documents of the ECHR and with models trained with the Wikipedia, combined with embeddings obtained from the Wikipedia, from the ECHR or from both.

approach	accuracy	precision	recall	F1
NEL on ground truth				
NN	.94	.48	.45	.45
NN+WE	.72	.25	.25	.25
NEL on automatic YAGO-level NERC				
NN	.69	.18	.15	.16
baselines				
Random	.51	.00	.00	.00
K-nn	.71	.14	.10	.10

Table 3: Results for Named Entity Linking on the test portion of the Wikipedia corpus.

NEs by a string-based representation that allows for better string overlap heuristics and a customized edit distance for abbreviation heuristics.

4. Argument Mining

In this section, we describe the annotation of a corpus to train Argument Mining tools. The corpus is composed of judgments of the European Court of Human Rights (ECHR) in English, obtained from the Court website[11]. This will allow us to compare our annotation to that of (Mochales Palau and Moens, 2009)[12].

We are currently working in a delimitation of the scope of annotation that provides a balance between descriptive adequacy and performance of analyzers. To approach that balance, we are analyzing inter-annotator agreement and also discrepancies between human and automated annotators, to identify concepts that produce inconsistencies and produce a more useful delimitation, in a cycle *training of annotators – annotation – analysis of discrepancies – refining of annotation guidelines*. We are currently undergoing extensive annotation of this corpus after a first iteration of this cycle.

[11] `hudoc.echr.coe.int`

[12] The dataset described in this paper is not available online.

4.1. Objectives of annotation of argumentative structure

The objective of our annotation is to identify arguments composed by claims and premises that are related to each other. Our annotation scheme is loosely based on (Toulmin, 2003), following the main adaptations that (Habernal, 2014) proposes to take the concepts from a theoretical model to practical annotation guidelines. Argument components are classified as *claims* or *premises*, with some genre-dependent attributes associated to each of these classes. The category of *major claim* is not distinguised in our annotation guidelines, as it was the main source of disagreement between annotators and it was not crucial for descriptive adequacy or application needs (Teruel et al., 2018).

The basic concepts of our annotation are:

Claim : a controversial statement whose acceptance depends on premises that support or attack it. Claims are the central components of an argument and they either support or attack the major claim. We associate each claim with the actor that has issued it.

Premise : they are the reasons given by the author for supporting or attacking the claims. They are not controversial but factual. Specifically for this corpus, We distinguish subclasses of Premises: Facts, Principles of Law and Case-law.

Argument components are connected to each other by relations, mainly *support* or *attack* relations (Simari and Rahwan, 2009). Claims support or attack other claims or a major claim, premises may support or attack claims or other premises. Additionally, we have established two more minor relations, specific for this corpus: *duplicate* (holding between claims or premises) and *citation* (holding between premises, when one cites a reference Case-law).

We have used brat (Stenetorp et al., 2012) as a tool for annotation. The guidelines for annotation, together with the annotated texts, are available at `https://github.com/PLN-FaMAF/ArgumentMiningECHR`.

4.2. Consistency of annotation, manual and automatic

For the first iteration of the cycle *training of annotators – annotation – analysis of discrepancies – refining of annotation guidelines*, four human annotators annotated 7 judgments from the ECHR, totaling 28,000 words. Approximately half of the words were annotated as belonging to an argument component.

We found a high agreement between annotators to determine whether a sentence contained an argument component, with Cohen's kappa ranging between $\kappa = .77$ and $\kappa = .84$. When this agreement is considered at token level, it varies between $\kappa = .59$ and $\kappa = .84$. We note that most disagreements occur between annotators that annotate less or more proportion of words as argumentative. Indeed, some annotators tend to consider more spans of text as argument components than others. However, there is a high agreement on spans identified as argumentative by annotators that consider less spans of text as argumentative. This has been addressed in the second version of the guidelines by a more application-oriented definition of argumentative text, focusing on an information retrieval scenario.

For the classification of argument components as premises or claims we found an agreement, ranging from $\kappa = .48$ to $\kappa = .51$ and from $\kappa = .56$ to $\kappa = .64$. We found that claims issued by the ECHR are a major source of disagreement, because the concept is mixed with that of fact or principle of law. This can be expected, as claims by a court in a judgment do have the status of principles of law after the judgment is issued, and principles of law have the same status as facts in a reasoning by a court. However, epistemologically these three concepts are difficult to reconcile. To a minor extent, claims issued by the government tend to be mixed with premises labeled as facts. Moreover, the category of premise as fact also accumulates a high number of disagreements with the category of non-argumentative text. There is also some confusion between premises interpreted as facts or as case-law, and also between premises considered case-law or law principles.

To assess the level of agreement for relations, we looked into relations that held between argument components where two annotators agreed. That meant between 46% and 74% of the components. For those, annotators agreed on the existence of a relation between components only in between 10% and 19% of the cases. When they agreed that a relation held between a given pair of components, annotators tended to agree on whether the relation was of attack, support or citation, with agreement ranging from 85% to 100% in most cases. However, the number of cases where such analysis could be carried out is so small that we require a bigger corpus to obtain more significant figures and draw conclusions upon them.

We also explored the relation between inter-annotator agreement and the performance of an automated classifier relying on the Argument classifier developed by (Eger et al., 2017), a neural end-to-end argumentation mining system with a multi-task learning setup. This system has been trained with part of the corpus, then annotated a different part of the corpus and its predictions compared with human annotations.

The comparison of human and automatic annotations is shown in Figure 1. We can see that the confusion between premises and non-argumentative text is higher than the confusion between claims and non-argumentative text, and the confusion between premises and non-argumentative text is also higher than the confusion between claims and non-argumentative text. In consequence, there seems to be a strong relation between disagreements between humans and misperformance of automatic analyzers. Addressing the first will probably have a very positive impact on the second. To address that, we have developed a refined version of the annotation guidelines, with more adequate and accurate definitions of concepts, and are currently working on annotating judgments with these guidelines.

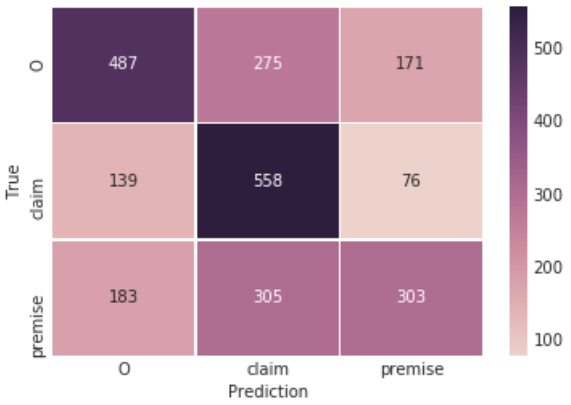

Figure 1: Confusion matrices for annotations of components between an automatic classifier and the human gold standard.

5. Summary of objectives and contributions

We have presented a work in progress for Named Entity Recognition, Classification and Linking and Argument Mining for the legal domain within the MIREL project. We have described our methodology to obtain a tool for NERC/NEL with little effort, and showed that results are promising. We have also described our approach to Argument Mining, where we are currently working on improving the annotation process to find a balance between descriptive adequacy and performance of analyzers. All tools and resources developed or in development are available at `https://github.com/PLN-FaMAF/legal-ontology-population` and `https://github.com/PLN-FaMAF/ArgumentMiningECHR`.

As future work we will improve the NERC/NEL by incorporating manually annotated examples from the ECHR, which has shown to produce good results. To optimize the annotation procedure, we will apply active learning techniques. We will also continue developing the corpus annotated for argument mining, to exploit it to train different kinds of learners, with a special focus on interpretability (i.e., Attention Networks (Cho et al., 2015)) and semi-supervised approaches (i.e., Ladder networks (Rasmus et al., 2015)).

6. Bibliographical References

Bengio, Y., Louradour, J., Collobert, R., and Weston, J. (2009). Curriculum learning. In Proceedings of the 26th Annual International Conference on Machine Learning, ICML '09, pages 41–48, New York, NY, USA. ACM.

Cardellino, C., Teruel, M., Alemany, L. A., and Villata, S. (2017). Legal NERC with ontologies, wikipedia and curriculum learning. In Mirella Lapata, et al., editors, Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics, EACL 2017, Valencia, Spain, April 3-7, 2017, Volume 2: Short Papers, pages 254–259. Association for Computational Linguistics.

Cho, K., Courville, A. C., and Bengio, Y. (2015). Describing multimedia content using attention-based encoder-decoder networks. CoRR, abs/1507.01053.

Eger, S., Daxenberger, J., and Gurevych, I. (2017). Neural end-to-end learning for computational argumentation mining. CoRR, abs/1704.06104.

Habernal, I., (2014). Argumentation in User-Generated Content: Annotation Guidelines. Ubiquitous Knowledge Processing Lab (UKP Lab) Computer Science Department, Technische Universität Darmstadt, April.

Hoekstra, R., Breuker, J., Bello, M. D., and Boer, A. (2007). The lkif core ontology of basic legal concepts. In Proceedings of the Workshop on Legal Ontologies and Artificial Intelligence Techniques (LOAIT 2007).

Linguistic Data Consortium. (2014). Deft ere annotation guidelines: Entities v1.7. http://nlp.cs.rpi.edu/kbp/2014/ereentity.pdf.

Lippi, M. and Torroni, P. (2016). Argumentation mining: State of the art and emerging trends. ACM Trans. Internet Techn., 16(2):10:1–10:25.

Mochales Palau, R. and Moens, M.-F. (2009). Argumentation mining: the detection, classification and structure of arguments in text. In Proceedings of the Twelfth International Conference on Artificial Intelligence and Law (ICAIL 2009), Twelfth international conference on artificial intelligence and law (ICAIL 2009), Barcelona, Spain, 8-12 June 2009, pages 98–109. ACM.

of Pisa, M. U. (2015). Wikiextractor. http://medialab.di.unipi.it/wiki/Wikipedia_Extractor.

Rasmus, A., Valpola, H., Honkala, M., Berglund, M., and Raiko, T. (2015). Semi-Supervised Learning with Ladder Networks. July.

Guillermo Ricardo Simari et al., editors. (2009). Argumentation in Artificial Intelligence. Springer.

Stanford NLP Group. (2016). Stanford named entity recognizer (ner). http://nlp.stanford.edu/software/CRF-NER.shtml.

Stenetorp, P., Pyysalo, S., Topić, G., Ohta, T., Ananiadou, S., and Tsujii, J. (2012). Brat: A web-based tool for nlp-assisted text annotation. In Proceedings of the Demonstrations at the 13th Conference of the European Chapter of the Association for Computational Linguistics, EACL '12, pages 102–107, Stroudsburg, PA, USA. Association for Computational Linguistics.

Suchanek, F. M., Kasneci, G., and Weikum, G. (2007). Yago: A core of semantic knowledge. In Proceedings of the 16th International Conference on World Wide Web, WWW '07, pages 697–706, New York, NY, USA. ACM.

Surdeanu, M., Nallapati, R., and Manning, C. D. (2010). Legal claim identification: Information extraction with hierarchically labeled data. In Proceedings of the LREC 2010 Workshop on the Semantic Processing of Legal Texts (SPLeT-2010), Malta, May.

Teruel, M. and Cardellino, C. (2017). n-domain or out-domain word embeddings? a study for legal cases. In Proceedings of the ESSLLI 2017 Student Session, Toulouse, France.

Teruel, M., Cardellino, F., Cardellino, C., and Villata, S. (2018). Increasing argument annotation reproducibility by using inter-annotator agreement to improve guidelines. In Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC'18). European Language Resources Association (ELRA), may.

Toulmin, S. E. (2003). The Uses of Argument. Cambridge University Press, July.

Winick, E. (2017). Lawyer-bots are shaking up jobs. MIT Technology Review, 12. https://www.technologyreview.com/s/609556/lawyer-bots-are-shaking-up-jobs/.