Claim Detection in Judgments of the EU Court of Justice

Marco Lippi[1⋆], Francesca Lagioia[2], Giuseppe Contissa[2,3], Giovanni Sartor[2,3], and Paolo Torroni[4]

[1] DISMI – Università degli Studi di Modena e Reggio Emilia
marco.lippi@unimore.it
[2] EUI – European University Institute
francesca.lagioia@eui.eu, giovanni.sartor@eui.eu
[3] CIRSFID – Università degli Studi di Bologna
giuseppe.contissa@unibo.it
[4] DISI – Università degli Studi di Bologna
paolo.torroni@unibo.it

Abstract. Mining arguments from text has recently become a hot topic in Artificial Intelligence. The legal domain offers an ideal scenario to apply novel techniques coming from machine learning and natural language processing, addressing this challenging task. Following recent approaches to argumentation mining in juridical documents, this paper presents two distinct contributions. The first one is a novel annotated corpus for argumentation mining in the legal domain, together with a set of annotation guidelines. The second one is the empirical evaluation of a recent machine learning method for claim detection in judgments. The method, which is based on Tree Kernels, has been been applied to context-independent claim detection in other genres such as Wikipedia articles and essays. Here we show that this method also provides a useful instrument in the legal domain, especially when used in combination with domain-specific information.

Keywords: Claim Detection, Argumentation Mining, Legal Arguments

1 Introduction

One of the most traditional yet lively research sub-areas at the intersection of Artificial Intelligence and Law is the study of argumentation in the legal context [5, 6]. Argumentation is a wide research field that spans across several different areas, having its roots in logic, philosophy, and linguistics, as it basically studies how different theses and opinions are proposed, debated and evaluated, taking into account their relations and inter-dependencies. The legal domain thus offers a natural scenario for the application of different argument models,

⋆ This work was done while Marco Lippi was at DISI – University of Bologna and Francesca Lagioia was at CIRSFID – University of Bologna.

in order to perform legal reasoning [26], to build specific ontologies [2], or to support the teaching of jurisprudence [3, 10].

From the Artificial Intelligence viewpoint, many contributions have been made in the context of building computational and logic models for legal arguments [26, 27], in case-based reasoning [1, 7], in the full semantic interpretation of judicial opinions [20], in yielding the syntactic structure of sentences using a rule-based parser in order to detect legal modifications [8], and also in the automatic extraction of arguments (or part thereof) from legal documents, as an application of the recent discipline of *argumentation mining* [21, 22]. Some recent works [36] are focused on the discovery and analysis of the internal structure of an arguments, the identification of its premises and the conclusions, and the internal syntactical and grammatical structure of each statement.

Building tools capable of automatically detecting arguments in legal texts would produce a dramatic impact on many disciplines related to Law, providing invaluable instruments for the retrieval of legal arguments from large corpora, for the summarization and classification of legal texts, and finally for the development of expert systems supporting lawyers and judges. Mochales Palau and Moens [22] gave an influential contribution in this domain, providing the first system for mining arguments from legal documents. Their system, specifically designed by experts in the legal domain, combines highly engineered feature construction, machine learning approaches, and a hand-crafted context-free grammar to infer links between arguments. Their results are yet hard to reproduce, since the dataset they used, made up of documents extracted from legal texts of the European Court of Human Rights (ECHR) and from the AraucariaDB, is currently not available, whereas their methodology exploits plenty of context-dependent information that was specifically extracted from that corpus.

In this work, we aim to contribute to the budding field of argumentation mining in legal documents by moving in two different directions. First, we present a novel, freely available, annotated corpus for argumentation mining in the legal context, accompanied by a set of guidelines that have been followed during the document labeling process. Second, we consider a machine learning approach for the extraction of claims from legal documents that has recently been applied to context-independent argument mining [18], and we evaluate it in this new genre. Our preliminary results show that context-independent claim detection is also helpful in the legal domain, thus providing a powerful framework that can be used in combination with domain-specific information.

2 Background

Argumentation mining is concerned with the automatic extraction of arguments from generic textual corpora. This has a self-evident application potential in a variety of domains. IBM recently funded a multi-million cognitive computing project called *Debater*, whose core technology is argumentation mining, and which aims to retrieve pro and con arguments concerning a given controversial topic.[5] If we consider user-generated content available on the Web, argumentation mining can be seen as the evolution of sentiment analysis: its goal is to detect not only opinions, but also the reasons behind them [15].

One of the pioneering applications of argumentation mining was in the legal domain. Building on Teufel's work on argumentation zoning [30], Hachey and Grover proposed a system for legal document summarization [16]. Mochales Palau and Moens [22] proposed the first argumentation mining system, focussing on the extraction of claims and their supporting premises from a collection of structured legal documents. They worked on two datasets: the European Court of Human Rights (ECHR) [21] and AraucariaDB, a collection maintained by the University of Dundee.[6] More recently, the Vaccine/Injury Project (V/IP) [4] was carried out, with the goal of extracting arguments from a set of judicial decisions involving vaccine regulations.

Although the general idea of argumentation mining is clear, a precise definition of the problem is difficult to obtain, because the problem is complex in itself (there are many sub-tasks that can be identified), and the very notion of argument is still a matter for discussion, and is somehow genre-dependent. An argument in law is different from an argument in an online discussion, although they do share commonalities.

A simple and intuitive characterization of an argument is given by Walton as a set of statements consisting of three parts: a conclusion, a set of premises, and an inference from the premises to the conclusion [33]. Aside from this basic premise/conclusion argument model, there are other noteworthy models are due to Toulmin [31] and Freeman [14]. A rather comprehensive account of argumentation models from an argument analysis perspective is given by Peldszus and Stede [25]. To add to the terminological complexity, in the literature conclusions are sometimes referred to as *claims*, premises are often called *evidence* or *reasons*, and the link between the two, i.e., the inference, is sometimes called the *argument* itself. The task of detecting the premises and conclusion of an argument, as found in a text of discourse, is typically referred to as *detection* or *identification* [33]. More specific sub-tasks are *claim detection* and *evidence detection* [17, 18, 28].

Even the targets of argumentation mining vary widely. Some research aims at extracting the arguments from generic unstructured documents, which is a fundamental step in practical applications [17], whereas other research starts from a given set of arguments and focuses on aspects such as the identification of attack/support [11] or entailment [9] relations between them, or on the classification of argument schemes [13] in the sense of Walton et al. [34]. The aforementioned approach by Mochales Palau and Moens [22] represents, to date, one of the few works whose goal was to implement a full-fledged argumentation mining system, albeit specific to a single genre.

---

[5] More about IBM Debating Technologies at `http://researcher.watson.ibm.com/researcher/view_group.php?id=5443`

[6] `http://corpora.aifdb.org/`

Diverse methodologies have been developed which involve aspects of natural language processing and understanding, information extraction, feature discovery and discourse analysis. In general, all the argument mining frameworks proposed so far can be described as multi-stage pipeline systems [19], whose input consists of natural, free text documents, and whose output is a marked-up document, where arguments (or parts of arguments, such as claims) are marked up. Each stage addresses a sub-task of the whole argumentation mining problem, by employing one or more machine learning and natural language processing methodologies and techniques.

A first stage usually consists in detecting which sentences in the input document are argumentative, which means that they contain an argument, or part thereof. Once argumentative sentences are singled out, one needs to detect the boundaries between the various argument components. Finally, a last stage in the pipeline considers these components in order to predict links between them and/or between the arguments they are part of. For an overview of the state of the art in argumentation mining, the reader can refer to [19]. In this work, we focus on the first stage of the pipeline, considering *concluding claims* as the target. By the concluding claim of an argument we mean its conclusion, namely, the proposition that is affirmed on the basis of the argument's premises. The concluding claims affirmed in the opinion conclusions are crucial for its understanding, as they highlight the core legal grounds supporting the judgment.

We choose to focus on judicial opinions since they are highly structured legal texts, in which an account of the arguments by different parties is first presented, and then the justification for the judge's decision is provided. Both aspects are extensively developed in the opinions of the European judges.

3 Corpus

The source corpus consists of fifteen relevant European Court of Justice (ECJ) decisions from 2001 to 2014 extracted from the EUR-LEX database, all related to data protection. These documents were manually labeled following a procedure that we describe in detail in the following subsections. These annotations will represent the *ground truth* for our claim detection system.

3.1 Data collection

This source was selected because: (a) ECJ decisions contain different types of legal arguments by different actors (e.g. arguments appealing to statues, principles or precedents, according to different interpretive canons, e.g. arguments by the parties, the Advocate General or the judges); (b) ECJ decisions have a standard (although not fixed) structure, in which the complex and highly variable structure of arguments is embedded; (c) the selected decisions come from the same domain (data protection), in which the annotators have some expertise. ECJ decision usually include the following sections:

1. preamble: information on the parties and the main object of the judgment;

2. legal context: listing of all the legal instruments used in the judgment;
3. background of the case: the procedural history of the case and the question referred to the court;
4. consideration on the question(s) referred: the observations submitted to the court by the parties and other actors such as the Governments of Member States, plus the responses of the Court;
5. costs: the attribution of costs;
6. ruling: the final decision and the orders to the parties.

3.2 Annotation guidelines

In analyzing the ECJ decisions, we did not consider Sections 1 (preamble), 2 (legal context) and 3 (background of the case), because they contain only legal and factual information, but no arguments are put forward. The most interesting part for our aims is Section 4 (consideration on the question(s) referred), which contains all argumentative steps leading to the final ruling. Section 5 (costs) was taken in to account. We did not consider Section 6 (ruling), since it usually repeats the top claims of Section 4, completed with orders to the parties. The text is divided into numbered paragraphs. In selecting arguments we proceeded as follows: for each paragraph, if arguments were present, we considered the chaining of arguments [32], identified the top-level argument, that is, the ultimate argument in the chain, and we annotated the claim corresponding to its conclusion, as well as the keywords signaling or introducing such argument. Highlighting keywords and markers in the text was useful for the purpose of keeping the annotations uniform.

In order to detect an argument we first considered the grammatical and syntactical structure of the text, looking for occurrences of conclusion indicators [12] such as "as a result", "therefore", "consequently", "thus", "for this reason". Nevertheless, sometimes the grammatical and the syntactical structures were not sufficient to detect arguments, and it was necessary to take into account the semantics and the legal context. For instance, consider the following statements taken from judgment C-301/06, paragraphs 28 and 38, respectively:

> *Ireland submits that* ***the choice of Article 95 EC as the legal basis for Directive 2006/24 is a fundamental error.***

> *Article 4 of Directive 2006/24 provides that the conditions for access to and processing of retained data must be defined by the Member States subject to the legal provisions of the Union and international law.*

We can say that the first statement introduces a claim that can be evaluated as the conclusion of an argument, while the second simply repeats the content of a legal provision and it is not part of an argumentative claim.

The contextual analysis also helped us to distinguish two uses of precedents and other legal sources: (1) an argumentative use, where the court refers to a precedent or source in order to reinforce and bolster its arguments supporting the decision; (2) a non-argumentative way, in which the court incidentally mentions a previous decision or an argument of other parties, as a contextual element. Only in the first case did we annotate the sentence as containing a claim. As an example, consider the following statements of the Court from judgment C-275/06, paragraph 36 and 62:

> *It should be recalled that* ***it is solely for the national court before which the dispute has been brought, and which must assume responsibility for the subsequent judicial decision, to determine in the light of the particular circumstances of the case both the need for a preliminary ruling in order to enable it to deliver judgment and the relevance of the questions which it submits to the Court*** *(Case C 217/05 Confederación Española de Empresarios de Estaciones de Servicio [2006] ECR I 11987, paragraph 16 and the case-law cited).*
>
> *It should be recalled that the fundamental right to property and the fundamental right to effective judicial protection constitute general principles of Community law (see respectively, to that effect, Joined Cases C 154/04 and C 155/04Alliance for Natural Health and Others [2005] ECR I 6451, paragraph 126 and the case-law cited, and Case C 432/05 Unibet [2007] ECR I 2271, paragraph 37 and the case-law cited).*

This first statement is an example of how the Court refers to a previous ruling to strengthen its argument, while the second one refers to a contextual element.

When a paragraph includes two or more top-level arguments, we highlighted the conclusions of both arguments and also marked the keywords or the expressions linking the arguments, such as “moreover”, “furthermore”, “additionally”, “it should be added that”, etc. As an example, consider the following two statements taken from judgment C-131/12, paragraph 22 in which we marked both the top-level claims, highlighted in bold in the text :

> *According to Google Spain and Google Inc.,* ***the activity of search engines cannot be regarded as processing of the data which appear on third parties web pages displayed in the list of search results.***
>
> *Furthermore, even if that activity must be classified as "data processing",* ***the operator of a search engine cannot be regarded as a "controller" in respect of that processing since it has no knowledge of those data and does not exercise control over the data****.*

When contiguous arguments are chained together, so that the preceeding ones are only meant to provide the premises for the last one, we have marked only the last argument (highlighted in bold in the text). Look at the following example:

> *As regards Article 12(b) of Directive 95/46 , the application of which is subject to the condition that the processing of personal data be incompatible with the directive, it should be recalled that, as has been noted in paragraph 72 of the present judgment, such incompatibility may result not only from the fact that*

> *such data are inaccurate but, in particular, also from the fact that they are inadequate, irrelevant or excessive in relation to the purposes of the processing, that they are not kept up to date, or that they are kept for longer than is necessary unless they are required to be kept for historical, statistical or scientific purposes.*
>
> *It follows from those requirements, laid down in Article 6(1)(c) to (e) of Directive 95/46, that even initially lawful processing of accurate data may, in the course of time, become incompatible with the directive where those data are no longer necessary in the light of the purposes for which they were collected or processed. That is so in particular where they appear to be inadequate, irrelevant or no longer relevant, or excessive in relation to those purposes and in the light of the time that has elapsed.*
>
> *Therefore, if it is found, following a request by the data subject pursuant to Article 12(b) of Directive 95/46, that the inclusion in the list of results displayed following a search made on the basis of his name of the links to web pages published lawfully by third parties and containing true information relating to him personally is, at this point in time, incompatible with Article 6(1)(c) to (e) of the directive because that information appears, having regard to all the circumstances of the case, to be inadequate, irrelevant or no longer relevant, or excessive in relation to the purposes of the processing at issue carried out by the operator of the search engine,* ***the information and links concerned in the list of results must be erased.***

When a party in the case mentions an argument by another to endorse that argument, we considered the conclusion of the argument as a claim by the party. As an example, look at the structure of the following statement, taken from judgment C-101/01, paragraph 31:

> *The Swedish Government submits that, when Directive 95/46 was implemented in national law, the Swedish legislature took the view that* ***processing of personal data by a natural person which consisted in publishing those data to an indeterminate number of people, for example through the internet, could not be described as "a purely personal or household activity" within the meaning of the second indent of Article 3(2) of Directive 95/46.*** *However, that Government does not rule out that the exception provided for in the first indent of that paragraph might cover cases in which a natural person publishes personal data on an internet page solely in the exercise of his freedom of expression and without any connection with a professional or commercial activity.*

In this example, the claim put forward by the Swedish legislature, introduced by the expression "took the view that", is endorsed by the Swedish Government, an endorsement expressed by the locution "submits that".

The annotated corpus is available at `http://argumentationmining.disi.unibo.it/aicol2015.html`.

4 Methods

In this work, we focus on the first stage of the pipeline sketched in Section 2, i.e., on argumentative sentence classification, and in particular on claim detection. Our goal is thus to detect sentences that contain a claim.

The most common approaches to this task typically employ machine learning systems, whose aim is to construct a classifier that is capable of associating a given sentence x with a label y that indicates whether or not the sentence contains a claim. There is a wide variety of methods for building such a classifier. They differ by their chosen machine learning algorithm and by how they represent sentences. Some techniques simply represent the sentence with the well-known bag-of-words model, in which a sentence x is just represented by the set of its words, regardless of their order, encoded into a linear vector. Advanced variants of that model also consider bigrams and trigrams of words. The most common existing approaches to claim detection rely on large sets of sophisticated features, that are very often domain-dependent and designed by hand to address the task of interest. While simple machine learning algorithms are typically used as off-the-shelf tools [19], a lot of effort is dedicated in these approaches to the development of such highly engineered features. This is the case, for example, in the work by Mochales Palau and Moens [22] on judicial decision, the works by the IBM Haifa research team in the context of the Debater project [17, 28], and the approach presented by Stab and Gurevych on persuasive essays [29]. Such works use, for example, the following inputs: pre-determined lists of special keywords that are usually highly indicative of the presence of an argument; the output of external classifiers that compute the sentiment or the subjectivity score of a sentence; semantic information coming from thesauri and ontologies like WordNet.

A recent work by Lippi and Torroni [18] has shown that the structure of a sentence is very often highly informative on the presence of argumentative components, such as claims. The key idea is that information coming from natural language processing, as in the case of parse trees, can be employed to measure similarity between sentences, and thus to detect fragments and structures that typically encode claims. As an example, consider the following two sentences:

> *The Netherlands Government submits that* ***Directive 95/46 does not preclude Member States from providing for greater protection in certain areas****.*

> *The Parliament also argues that* ***reliance on Article 95 EC as the legal basis is not invalidated by the importance attributed to combating crime****.*

The first sentence is taken from judgment C-101/01 (paragraph 93), while the second one is taken from judgment C-301/06 (paragraph 37) and they have both been labeled as containing a claim (highlighted in bold) in our corpus. The parse trees for these sentences are shown in Figure 1, where boxed nodes highlight the common structures, in this case consisting of a subordinate introduced by a third-person verb (VBZ) and the preposition “that”. The two verbs introducing such

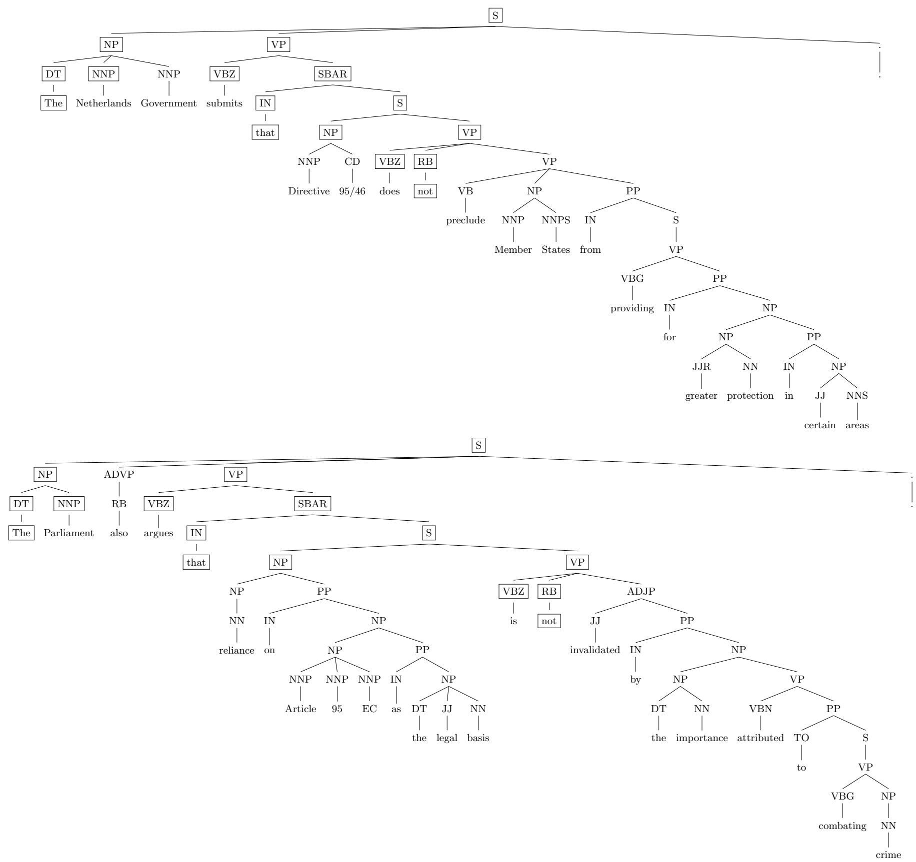

Fig. 1. Constituency trees for two sentences containing claims. Boxed nodes highlight the common structure of a subordinate introduced by a third-person verb (VBZ) and the *that* preposition (IN). Examples are taken from judgments collected in our corpus.

subordinates (submit, argue) are also indicative of the presence of an argument. Other patterns are frequently observed in sentences containing claims.

The structure of a sentence is thus highly indicative of the presence of a claim, and constituency parse trees represent a very powerful instrument to capture such information. Based on this observation, Lippi and Torroni [18] built a claim detection system that employs a Support Vector Machine (SVM) classifier capturing similarities between parse trees through Tree Kernels [23]. The Tree Kernel approach has been shown to outperform competitors exploiting classic, handcrafted features, widely used in NLP, such as bag-of-words, bigrams, trigrams, part-of-speech tags and lemmas, while achieving results comparable to highly sophisticated systems, specifically designed for context-dependent claim detection.

Kernel methods, and in particular Tree Kernels, have a quite long tradition in natural language processing applications, including relation extraction, named entity recognition, or question classification [24]. A kernel machine classifier learns a function $f : \mathcal{X} \to \mathcal{Y}$ where $\mathcal{X}$ is the input space, usually a vector space representing features, and $\mathcal{Y}$ is the output space representing the set of labels, or categories, to be distinguished (in our case, claim vs. other). To learn function f, a loss function is minimized over a set of N given observations, which is a dataset $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^{N}$. Examples $x_i \in \mathcal{X}$ are not necessarily represented by vectors of features, but they can also exploit structured data, in order to encode relational information, as it happens with trees or graphs. A Tree Kernel (TK) can be basically thought of as a *similarity measure* between two trees, that evaluates the number of their common substructures, sometimes also called *fragments*. According to the definition of fragments, different TK functions can be constructed.

For example, one could consider only complete subtrees as allowed fragments, as well as define more complex fragment structures. Intuitively, each possible tree fragment is associated with a different feature in a high-dimensional vectorial space, where the j-th feature simply counts the number of occurrences of the j-th tree fragment: the TK can therefore be computed as the dot product between two such representations of different trees. A kernel machine is then defined, which exploits the structured information encoded by the tree kernel function $K(x, z)$:

$$f(x) = \sum_{i=1}^{N} \alpha_i y_i \phi(x_i) \cdot \phi(x) = \sum_{i=1}^{N} \alpha_i y_i K(x_i, x) \tag{1}$$

where ϕ is the feature mapping induced by the tree kernel K, and N is the number of support vectors. In general, the kernel between two trees T_x and T_z can be computed as:

$$K(T_x, T_z) = \sum_{n_x \in N_{T_x}} \sum_{n_z \in N_{T_z}} \Delta(n_x, n_z) \tag{2}$$

where N_{T_x} and N_{T_z} are the set of nodes of the two trees, and $\Delta(\cdot, \cdot)$ measures the score between two nodes, according to the definition of the considered fragments.

In this work we consider the Partial Tree Kernel (PTK) [23], which allows the most general set of fragments (called Partial Trees), being any possible portion of subtree at the considered node. The higher the number of common fragments, the higher the score Δ between two nodes.

The representation power behind tree kernels is evident. Basically, a kernel-like PTK is capable of automatically generating a very rich feature set, that captures structured representations without the need of a costly, hand-crafted feature engineering process. Nevertheless, it is very interesting to notice that the TK framework allows us to include in the representation of each example also a plain vector of features, which can enrich the description of the considered instance given by the parse tree. In this case, the final kernel would be computed as the combination between a classic kernel between feature vectors K_V and the kernel between trees K_T, e.g., with a weighted sum or product of the two contributions. Also note that the use of a TK by itself does not take into account information about the context of the considered documents, which, in our case, is the legal domain, nor the whole structure of the document. For this reason, context-dependent information could be appended to the feature vector to be combined with the TK, thus exploiting both context-dependent and context-independent information, whereas relational learning algorithms could be applied in order to capture the relations across different sentences.

5 Results

For our experiments, we employed the new corpus of judgments of the ECJ related to data protection described in Section 3. We employed a leave-one-out procedure (LOO), as customary in machine learning experimental evaluation. The procedure dictates that in turn, each judgment be considered as a test case, while all the other documents constitute the training and validation sets. With N documents, training and testing are thus independently performed N times, and results are finally averaged across all the runs.

We used the Stanford CoreNLP suite[7] both to split each document into sentences, and to compute the parse tree for each sentence. We obtained in this way a total of 1,096 sentences, of which 435 were labeled as positive (i.e., containing a claim) and the remaining 661 as negative (i.e., not containing any claim). Aside from computing the parse trees, we also extracted from each sentence a vector of features, that have been extensively used in a variety of Natural Language Processing applications: bag-of-words, bag-of-bigrams and bag-of-trigrams for words, stems and part-of-speech tags.

We thus trained three distinct classifiers: (1) an SVM based on the PTK described in Section 4 as the kernel computed over the parse trees (we exploited a combination of both stemmed and not-stemmed constituency parse trees); (2) an SVM trained with a linear kernel over the vector of features only; (3) an SVM combining (summing) the PTK and the linear kernel over the feature vector. We will refer to these three classifiers as PTK, FV, and PTK+FV. For all classifiers, we selected the SVM regularization parameter C by employing three documents as a validation set. We relied on default values for the other PTK parameters.

Table 1 shows the results obtained by the three classifiers with the LOO procedure, macro-averaged on the 15 test documents, and also the results obtained with a random baseline classifier. Since our problem is a binary classification task (with two classes only), we define as True Positives (TP) the correctly detected elements of the positive class, as False Positives (FP) the negative examples that are wrongly classified as positives, and as False Negatives (FN) the positive cases that are not retrieved. Standard classification measurements for these kind of tasks include Precision ($P = \frac{TP}{TP+FP}$), Recall ($R = \frac{TP}{TP+FN}$), and $F_1 = \frac{2PR}{P+R}$, i.e., the harmonic mean between Precision and Recall.

[7] `http://nlp.stanford.edu/software/corenlp.shtml`

Classifier	P	R	F_1
Random	39.5	39.5	39.5
FV	40.0	63.1	47.9
PTK	39.7	80.5	52.4
PTK + FV	44.3	78.3	55.4

Table 1. Results obtained on our ECJ corpus, macro-averaged over the 15 documents.

The results show that all approaches perform much better than a random predictor, even with a relatively small training set. It is interesting to note that the combination of PTK and FV achieves the best performance, thus indicating that the information exploited by the two distinct approaches is somehow complementary, and that PTK could be conveniently used also in combination with more context-dependent information. In order to assess the statistical significance of these results, we run a Wilcoxon paired test [35] on the F_1 values obtained on each document, which produced a p-value < 0.01 for the PTK+FV classifier with respect to FV.

Finally, consider also that the approaches used here do not take into account the whole document structure, which is instead a crucial piece of knowledge for retrieving the concluding claims, as explained by the guidelines illustrated in Section 3. Therefore, it is clear that there is a large margin of improvement for the considered task, especially if including in the model contextual and relational information.

6 Conclusion

The ECJ decisions related to data protection are a small-sized but novel annotated corpus for argumentation mining in the legal domain. We are actively working on its extension. Nevertheless, we hope that it can represent a useful benchmark for future work in this domain. We have demonstrated that a context-independent methods such as the Tree Kernel-based classifier proposed in [18] could be a valuable asset for claim detection in this genre, especially when used in combination with domain-specific information. We are aware that this is only a first step, and there is certainly room for improvement. In the future, we plan to expand our corpus, and to extend the analysis to the labeling and prediction of premises, the labeling and prediction of support/attack links, and the extraction of argument maps directly from text. One important remark is that labeling was made using a considerable amount of information from the document and discourse structure. Not only ECJ decisions are structured in well-defined sections, only some of which contain argumentative content, but the structure itself of the argumentation within each section was analyzed and captured by the used labeling (top-level claims, embedded arguments, arguments referring to other arguments, such as strengthening arguments or repeated arguments, etc.). Often, the labeler is able to correctly identify a concluding claim thanks to the structure of the argumentation. By contrast, our classifier—which is nevertheless able to provide acceptable results—does not use this structure. This opens an avenue for further work that should go in the direction of exploiting the discourse structure. Statistical relational learning could thus represent a perfectly suitable framework for exploiting relational information across sentences. Another key contribution could also come from deep learning, which has recently achieved breakthrough results in a variety of tasks related to natural language processing.

In future work, we will also compare our approach with other classifiers and in particular with a simple rule-based one, that implements basic pattern recognition rules. We expect that our system and the classifier using pattern recognition rules will deliver similar outcomes with regard to small source corpora, but that our approach will deliver better results when applied to larger sets of documents.

References

1. Aleven, V.: Using background knowledge in case-based legal reasoning: a computational model and an intelligent learning environment. Artificial Intelligence 150(1), 183–237 (2003)
2. Alexander, B.: Lkif core: Principled ontology development for the legal domain. Law, ontologies and the semantic web: channelling the legal information flood 188, 21 (2009)
3. Ashley, K.D., Desai, R., Levine, J.M.: Teaching case-based argumentation concepts using dialectic arguments vs. didactic explanations. In: Intelligent Tutoring Systems. pp. 585–595. Springer (2002)
4. Ashley, K.D., Walker, V.R.: Toward constructing evidence-based legal arguments using legal decision documents and machine learning. In: Francesconi, E., Verheij, B. (eds.) ICAIL 2012, Rome, Italy. pp. 176–180. ACM (2013)
5. Bench-Capon, T., Freeman, J.B., Hohmann, H., Prakken, H.: Computational models, argumentation theories and legal practice. In: Reed, C., Norman, T. (eds.) Argumentation Machines, Argumentation Library, vol. 9, pp. 85–120. Springer Netherlands (2004)
6. Bench-Capon, T., Prakken, H., Sartor, G.: Argumentation in legal reasoning. In: Simari, G., Rahwan, I. (eds.) Argumentation in Artificial Intelligence, pp. 363–382. Springer US (2009)
7. Bench-Capon, T., Sartor, G.: A model of legal reasoning with cases incorporating theories and values. Artificial Intelligence 150(1), 97–143 (2003)
8. Brighi, R., Lesmo, L., Mazzei, A., Palmirani, M., Radicioni, D.P.: Towards semantic interpretation of legal modifications through deep syntactic analysis. In: Proceedings of the 2008 conference on Legal Knowledge and Information Systems: JURIX 2008: The Twenty-First Annual Conference. pp. 202–206. IOS Press (2008)
9. Cabrio, E., Villata, S.: A natural language bipolar argumentation approach to support users in online debate interactions. Argument & Computation 4(3), 209–230 (2013)
10. Carr, C.S.: Using computer supported argument visualization to teach legal argumentation. In: Visualizing argumentation, pp. 75–96. Springer (2003)
11. Chesñevar, C.I., McGinnis, J., Modgil, S., Rahwan, I., Reed, C., Simari, G.R., South, M., Vreeswijk, G., Willmott, S.: Towards an argument interchange format. The Knowledge Engineering Review 21(4), 293–316 (2006)

12. Copi, I.M., Cohen, C., McMahon, K.: Introduction to Logic: Pearson New International Edition. Pearson Higher Ed (2013)
13. Feng, V.W., Hirst, G.: Classifying arguments by scheme. In: Lin, D., Matsumoto, Y., Mihalcea, R. (eds.) The 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, Proceedings of the Conference, 19-24 June, 2011, Portland, Oregon, USA. pp. 987–996. ACL (2011)
14. Freeman, J.B.: Dialectics and the macrostructure of arguments: A theory of argument structure, vol. 10. Walter de Gruyter (1991)
15. Habernal, I., Eckle-Kohler, J., Gurevych, I.: Argumentation mining on the web from information seeking perspective. In: Cabrio, E., Villata, S., Wyner, A. (eds.) Proceedings of the Workshop on Frontiers and Connections between Argumentation Theory and Natural Language Processing. Forlì-Cesena, Italy, July 21-25, 2014. CEUR Workshop Proceedings, vol. 1341. CEUR-WS.org (2014)
16. Hachey, B., Grover, C.: Extractive summarisation of legal texts. Artificial Intelligence and Law 14(4), 305–345 (2006)
17. Levy, R., Bilu, Y., Hershcovich, D., Aharoni, E., Slonim, N.: Context dependent claim detection. In: Hajic, J., Tsujii, J. (eds.) COLING 2014, Dublin, Ireland. pp. 1489–1500. ACL (2014)
18. Lippi, M., Torroni, P.: Context-independent claim detection for argument mining. In: Yang, Q., Wooldridge, M. (eds.) Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence, IJCAI 2015, Buenos Aires, Argentina, July 25-31, 2015. pp. 185–191. AAAI Press (2015)
19. Lippi, M., Torroni, P.: Argumentation mining: State of the art and emerging trends. ACM Transactions on Internet Technology 16(2), 10:1–10:25 (Mar 2016)
20. McCarty, L.T.: Deep semantic interpretations of legal texts. In: Proceedings of the 11th international conference on Artificial intelligence and law. pp. 217–224. ACM (2007)
21. Mochales Palau, R., Ieven, A.: Creating an argumentation corpus: do theories apply to real arguments? A case study on the legal argumentation of the ECHR. In: Proceedings of the Twelfth International Conference on Artificial Intelligence and Law (ICAIL 2009), Barcelona, Spain, 8-12 June 2009. pp. 21–30. ACM (2009)
22. Mochales Palau, R., Moens, M.F.: Argumentation mining. Artificial Intelligence and Law 19(1), 1–22 (2011)
23. Moschitti, A.: Efficient convolution kernels for dependency and constituent syntactic trees. In: Frnkranz, J., Scheffer, T., Spiliopoulou, M. (eds.) Machine Learning: ECML 2006, LNCS, vol. 4212, pp. 318–329. Springer Berlin Heidelberg (2006)
24. Moschitti, A.: State-of-the-art kernels for natural language processing. In: Tutorial Abstracts of ACL 2012. pp. 2–2. ACL '12, Association for Computational Linguistics, Stroudsburg, PA, USA (2012)
25. Peldszus, A., Stede, M.: From argument diagrams to argumentation mining in texts: A survey. International Journal of Cognitive Informatics and Natural Intelligence (IJCINI) 7(1), 1–31 (2013)
26. Prakken, H., Sartor, G.: A dialectical model of assessing conflicting arguments in legal reasoning. In: Logical Models of Legal Argumentation, pp. 175–211. Springer (1997)
27. Prakken, H., Sartor, G.: The role of logic in computational models of legal argument: a critical survey. In: Computational logic: Logic programming and beyond, pp. 342–381. Springer (2002)
28. Rinott, R., Dankin, L., Perez, C.A., Khapra, M.M., Aharoni, E., Slonim, N.: Show me your evidence - an automatic method for context dependent evidence detection. In: Màrquez, L., Callison-Burch, C., Su, J., Pighin, D., Marton, Y. (eds.)

Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, EMNLP 2015, Lisbon, Portugal, September 17-21, 2015. pp. 440–450. The Association for Computational Linguistics (2015)

29. Stab, C., Gurevych, I.: Identifying argumentative discourse structures in persuasive essays. In: Moschitti, A., Pang, B., Daelemans, W. (eds.) EMNLP 2014, Doha, Qatar. pp. 46–56. ACL (2014)
30. Teufel, S.: Argumentative zoning. PhD Thesis, University of Edinburgh (1999)
31. Toulmin, S.E.: The Uses of Argument. Cambridge University Press (1958)
32. Walton, D.: Fundamentals of Critical Argumentation. Critical Reasoning and Argumentation, Cambridge University Press (2006)
33. Walton, D.: Argumentation theory: A very short introduction. In: Simari, G., Rahwan, I. (eds.) Argumentation in Artificial Intelligence, pp. 1–22. Springer US (2009)
34. Walton, D., Reed, C., Macagno, F.: Argumentation Schemes. Cambridge University Press (2008)
35. Wilcoxon, F.: Individual comparisons by ranking methods. Biometrics bulletin 1(6), 80–83 (1945)
36. Wyner, A., van Engers, T.: From argument in natural language to formalised argumentation: Components, prospects and problems. Proceedings of the Worskhop on Natural Language Engineering of Legal Argumentation, Barcelona, Spain (2009)