A Decade of Legal Argumentation Mining: Datasets and Approaches

Gechuan Zhang[1][0000−0002−8818−8540], Paul Nulty[2][0000−0002−7214−4666], and David Lillis[1][0000−0002−5702−4463]

[1] School of Computer Science, University College Dublin, Ireland
`gechuan.zhang@ucdconnect.ie`, `david.lillis@ucd.ie`
[2] Department of Computer Science and Information Systems, Birkbeck, University of London, UK
`p.nulty@bbk.ac.uk`

Abstract. The growing research field of argumentation mining (AM) in the past ten years has made it a popular topic in Natural Language Processing. However, there are still limited studies focusing on AM in the context of legal text (Legal AM), despite the fact that legal text analysis more generally has received much attention as an interdisciplinary field of traditional humanities and data science. The goal of this work is to provide a critical data-driven analysis of the current situation in Legal AM. After outlining the background of this topic, we explore the availability of annotated datasets and the mechanisms by which these are created. This includes a discussion of how arguments and their relationships can be modelled, as well as a number of different approaches to divide the overall Legal AM task into constituent sub-tasks. Finally we review the dominant approaches that have been applied to this task in the past decade, and outline some future directions for Legal AM research.

Keywords: Argumentation Mining · Legal Text · Text Analysis.

1 Introduction

Since Mochales and Moens presented their work on detecting arguments from legal texts in 2011, argumentation mining (AM), automatic detection of arguments and reasoning from texts [13], has become a popular research field. Meanwhile, attention in legal text processing has grown both in research and industry, leading to progress in new tasks such as legal topic classification [14], judicial decision prediction [4], and Legal AM [11]. Given that arguments are a core component of legal analysis, Legal AM has many important potential applications.

Although there are some works that describe the state-of-the-art of artificial intelligence (AI) and law [2], which have introduced AM, there is still a lack of a thorough review of Legal AM and its datasets or tools. Here, we present what is to our knowledge the first survey of Legal AM from a data-driven perspective. In particular, our work reviews this interdisciplinary field from two aspects: 1) corpus annotation, 2) argument extraction and relation prediction.

The lack of suitable open-source corpora is still a challenge in Legal AM, and complex annotation schemes can make evaluation difficult. Most present Legal AM work focuses on detecting text arguments, since relation prediction is the remaining challenge. In the remainder, Section 2 provides the related background in computational argumentation as well as the models used to structure human language. Section 3 discusses the existing annotation schemes and corpus creation. Section 4 investigates practical methods and the implementation of argument extraction and relation prediction in legal text. Section 5 contains our conclusions and prospects for Legal AM in the future.

2   Related Work

2.1   Computational Argumentation

In order to detect text arguments automatically, computational argumentation that expresses human language into structured data is required. At present, there are two types of computational argumentation models: abstract argumentation models (aka. argumentation frameworks [6], AFs), and structural argumentation models [22, 30], which individually focus on a macro (external) or a micro (internal) structure of argumentation. Abstract argumentation models treat each argument as the elementary unit without further details and emphasise relationships between arguments. To deal with the complex linguistic environment of legal texts, inner argumentation structure is required. As a result, structural argumentation models, including components within individual argument, are often used in Legal AM annotation scheme.

2.2   Structural Argumentation Model

Structural argumentation models assume a tentative proof of a given argument, then apply a set of rules on their substructures in order to formalise it and represent internal argument components and relations [9] . The logic-based definition of argument in structural argumentation models presents as a pair $< \phi, \alpha >$, where ϕ is a set of support formulae, and α is the consequent [1]. Here, we review two classic structural argumentation models.

- **Toulmin Model** [22] is a classic argumentation model that considers the inner structure of arguments. It has been used in debates, persuasive articles, and academic writing, long before being applied in NLP tasks. [22] designs a complete argument structure consisting of six components: *claim* (*conclusion*), *ground* (*data*), *warrant*, *support*, *qualifier*, *rebuttal*. The first three are the foundations which every argument starts with.
- **Walton Model** [30] proposes a simplified structure. [30] states an argument as a set of statements (propositions), made up of three components: a *conclusion*, a set of *premises*, and an *inference* from the premises to the conclusion. The model also includes higher-level bipolar relations between arguments: an argument can both be *supported* or *attacked* by other arguments.

3   Creating Annotated Legal Corpora

Like most interdisciplinary studies, the requirement of professional guidance increases the cost in time and labour when developing new AM corpora [10, 18]. This situation leads to two urgent needs in Legal AM: first, legal text corpora with accurate manual annotation; second, basic standard protocols when creating annotations. This section reviews several important works that create legal argument annotation schemes and describe the annotation of various types of legal texts, including case laws [11, 32], online comments on public rules [16], and judicial decisions [26]. The papers and corpora discussed in this work are listed in Table 1. Annotation details are concluded in Table 2. This work focuses on English texts. Legal texts in other languages [23] are also worth exploring in the future study of Legal AM.

Table 1. Papers on Argumentation Mining on Legal Text (ECHR=European Court of Human Rights, CDCP=Consumer Debt Collection Practices, VICP=Vaccine Injury Compensation Program, CanLII=Canadian Legal Information Institute, CA=corpus annotation, AD=argument and relation detection, doc=document, set=sentence, rec=record).

Authors	Abbr.	Source	Task	Corpus Size
Mochales and Moens [13]	MM2011		CA, AD	47 doc, 2,571 set
Teruel et al. [21]	TCCA2018	ECHR	CA	7 doc
Poudyal et al. [18]	PSI2020		CA, AD	42 doc
Niculae et al. [15]	NPC2017		CA, AD	731 rec, 3,800 set
Park and Cardie [17]	PC2018	CDCP	CA	731 rec, 3,800 set
Galassi et al. [7]	GLT2021		AD	
Walker et al. [26]	WCDL2011	VICP	CA	30 doc
Grabmair et al. [8]	GACS2015		AD	
Walker et al. [27]	WHNY2017		CA	20doc, 5,674 set
Walker et al. [24]	WFPR2018		CA	30doc, 8,149 set
Walker et al. [28]	WPDL2019	BVA	CA, AD	50doc, 6,153 set
Westermann et al. [31]	WSWA2019		AD	
Walker et al. [29]	WSW2020		CA, AD	75 doc, 623 set
Xu et al. [32]	XSA2020		CA, AD	683 doc, 30,374 set
Xu et al. [34]	XSA2021a	CanLII	CA, AD	1,148 doc, 127,330 set
Xu et al. [33]	XSA2021b		CA, AD	2,098 doc, 226,576 set

[12] provided the initial study on computational argumentation in legal text. In MM2011, they produced a corpus including 47 English-language cases (judgments and decisions) from the HUDOC[3] open-source database of the European Court of Human Rights (ECHR), a common resource for legal text processing research. MM2011 applied a sentence-level annotation scheme on ECHR files based on Walton's model. Segmented clauses were labelled as: *premise*, *conclusion* and *non-argumentative*. According to the distribution of clause-types in MM2011, there was an imbalance between premises and conclusions. [13] suggested one conclusion was often connected with multiple premises to build up a complete and stable argument in practical legal files. The annotation scheme

---

[3] https://hudoc.echr.coe.int/eng

in MM2011 had two further aspects. First, it considered the recurrent structure of sub-arguments in an argument. MM2011 concluded the argumentation into a tree structure where the leaves were arguments linked through argument relations, and all together supported a final conclusion. Second, the argument relations were annotated as rhetorical patterns. MM2011 explained that they did not judge the interaction between rhetorical and argument relations. The final IAA between four lawyers reached 0.75 of Cohen's κ [5].

Table 2. Legal Text Annotation Result (LA=logic annotation of argumentation model, CA=character annotation of legal context, Cmp=component, Rel=relation, IR=inner relation, OR=outer relation, Bi=bipolar relation, IRA=implicit relation annotation, ERA=explicit relation annotation; a=argument, c=component, r=relation, s=summary, f=full text, cκ=Cohen's κ, α=Krippendorf's α, N/A=not applicable).

Paper	LA	CA	Cmp	Rel	IR	OR	Bi	IRA	ERA	IAA
MM2011	*	*	*	*	*	*	*		*	a 0.75 (cκ)
TCCA2018	*	*	*	*	*	*	*		*	a 0.77-0.84 (cκ) c 0.48-0.64 (cκ) r 0.85-1.00 (cκ)
PSI2020	*		*	*	*			*		a 0.80 (cκ)
NPC2017	*	*	*	*	*				*	c 0.65 (α)
PC2018	*	*	*	*	*				*	r 0.44 (α)
WCDL2011	*	*	*	*	*		*		*	N/A
WFPR2018	*	*	*	*	*		*		*	N/A
WPDL2019	*	*	*	*	*		*		*	N/A
WSW2020	*	*	*	*	*		*		*	N/A
XSA2020	*		*							c (s/f) 0.71/0.77 (cκ)
XSA2021a	*		*							c (s/f) 0.71/0.83 (cκ)
XSA2021b	*		*							c (s/f) 0.73/0.60 (cκ)

Although MM2011 did not open-source the data, another ECHR AM corpus was more recently released by PSI2020. PSI2020 used the same corpus annotation process as MM2011. Four annotators achieved Cohen's κ inter-annotator agreement (IAA) of 0.80. For each clause, PSI2020's annotation included: a unique identifier and a character offset for start and end. Clause types in PSI2020 was aligned with MM2011: *premise*, *conclusion*, and *non-argument*. The PSI2020 annotation scheme highlighted the overlap between arguments: some clauses may be both premises and conclusions for different arguments. PSI2020 stored two types of information for each argument: 1) a list of clauses annotated as premises, and 2) the unique conclusion clause of the argument. The conclusion clause in each argument was treated as the conclusion type, any clause in the premise list was a premise type, and a clause which does not appear in any argument was a non-argument type. Unlike MM2011, PSI2020 omitted relations between individual arguments. The support relations from premises to conclusions were not explicitly annotated with labels. Instead, PSI2020 stored whole arguments as items and implicitly presented the support relations among each argument.

TCCA2018 includes annotations of 7 ECHR judgments (28,000 words). Their annotation scheme merged both the Toulmin model and previous guidelines [20] into three types of argument components: *major claim*, *claim* and *premise*. In

contrast to the premise/conclusion model in MM2011 and PSI2020, TCCA2018 treated the *major claim* as the highest level that can be *supported* or *attacked* by other arguments [20]. The bipolar relations between premises and claims also differ from the implicit support connections in PSI2020. Moreover, TCCA2018 conducted further classification on claims and premises: each (major-) claim was associated with its actor (ECHR, applicant, government), and premises were classified with sub-labels (Facts, Principles of Law and Case Law). TCCA2018 annotate both *support* and *attack* relations between argument components. In addition, TCCA2018 established two minor argument relations: *duplicate*, and *citation*. One of the seven judgements was annotated by all 4 annotators as training material (Cohen's $\kappa \geq 0.54$). TCCA2018 suggested IAA on argumentative/non-argumentative sentences was high (κ ranging between 0.77 and 0.84). The IAA dropped when annotating argument components, mainly due to disagreements of major claims.

Another widely used Legal AM corpus, Consumer Debt Collection[4] Practice (CDCP), is annotated by PC2018. The data consists of 731 user comments on Consumer Debt Collection Practices (CDCP) rules by the Consumer Financial Protection Bureau (CFPB). In order to structure the arguments, PC2018 uses a self-designed annotation scheme containing two parts: elementary units and support relations. The elementary units are sentences or clauses with different semantic types. Non-argumentative parts in comment texts (i.e., greetings, names) were removed when segmenting. To evaluate arguments, PC2018's annotations include two types of support relations: reason and evidence.

Apart from ECHR cases, the Research Laboratory for Law, Logic and Technology[5] (LLT Lab) from Hofstra University has annotated diverse samples of judicial decisions from U.S. courts. Their Vaccine/Injury Project (V/IP) used rule-based protocols, Default-Logic Framework (DLF) [25], to extract arguments in judicial decisions selected from the U.S. Court of Federal Claims. WCDL2011 modelled the fact-finding reasoning (a special argumentation in law) with DLF annotations. The annotation process (extracting the DLF structure from the judicial decision) is two-step. First, identifying sentences including argumentation information. Second, annotating sentences' inferential roles and support-levels in the rule-tree. WCDL2011 designed logical connectives [26] to represent argumentation relations between supporting reasons (premises) and conclusions. In addition, evidentiary propositions (premises and conclusions) have plausibility-values to measure the level of confidence in legal argumentation. The complexity of DLF made the manual annotation much harder. As a result, the final V/IP corpus in WCDL2011 contained sufficient semantic and logic information, which is represented in a rule-tree structure and stored in XML files.

The Veteran's Claim Dataset (or BVA) is another publicly available corpus annotated by the LLT Lab, using judicial-claim decisions from the Board of Veterans Appeals (BVA). WHNY2017 regarded legal argumentation the same as legal reasoning, and also modelled arguments with premise/conclusion model. The

---

[4] http://www.regulationroom.org/
[5] https://www.lltlab.org/

BVA decisions were annotated with semantic information of legal professional argumentation, including sentence roles and propositional connectives. These two groups of annotations matched the components and relations in broad argumentation model. The annotation scheme in WHNY2017 involved ten sentence reasoning roles and eight propositional connectives. The sentence types then acted as anchors when mining arguments (in WSWA2019). The propositional connectives represented argumentation relations from premises to conclusions. The argumentation relations have two properties: polarity and logical functionality. The polarity defines the support/oppose relation between the premises and the conclusion. The functionality measures the plausibility of an argument. The annotation work on BVA datasets continued for years; the initial corpus in WHNY2017 was only 20 documents (5,674 sentences), which was later enlarged to 30 documents (8,149 sentences) in WFPR2018. WPDL2019 expanded the dataset and analysed 50 judicial decisions. In the recent WSW2020, a second BVA dataset (25 decisions) has been annotated and published.

In a similar vein, the Intelligent Systems Program from University of Pittsburgh developed a series of corpora based on legal cases, which were sampled from the Canadian Legal Information Institute[6] (CanLII). They annotated argument structure as the *legal argument triples* (*IRC triples*): 1) *issue*, the legal question addressed in a legal case; 2) *conclusion*, the court's decision for the issue; 3) *reason*, sentences of why the court reached the conclusion. Based on the IRC annotation scheme, two annotators identified sentence-level argument components that form pairs of human-prepared summaries and full texts cases. They conduct annotations in two steps: first, annotating the case summaries in terms of IRC triples; second, annotating the corresponding sentences in full texts by mapping the annotations from summaries. This Legal AM dataset is still under development. From its initial version in XSA2020 with 574 legal case summaries and 109 full texts, the research group have enlarged the number of annotated documents to 574 full texts in XSA2021a. The latest CanLII corpus in XSA2021b contains 1049 annotated pairs of legal case summaries and full texts.

4   Practical Approaches for Legal Argumentation Mining

AM systems are generally organised as a two-stage pipeline: *argument extraction* and *relation prediction* [3]. Argument extraction, which typically contains sub-tasks, aims to identify arguments from input texts. Relation prediction focuses on the relations between (or within) identified arguments. Although identifying accurate argument boundaries is always a problem under discussion [19], AM annotations on legal text are usually at the sentence level, where a complete argument is a group of sentences or clauses with different logic functions.

After analysing the literature, we divide the Legal AM problem into the following sub-tasks: 1) argument information detection, 2) argument component classification, 3) argument relation prediction. Argument information detection

---

[6] https://www.canlii.org/en/

and argument component classification together comprise argument extraction. Table 3 summarises the prominent papers in the field, under a number of headings. It includes a) the specific sub-tasks that each study attempted to solve, b) the particular technologies used for AM and c) the argument components that the reasoning was performed on, along with the specific types of relationship that could exist between argument components in the model that was used.

Table 3. Legal Text Annotation Schemes and Analysis Technologies (ID=Information Detection, CC=Component Classification, RP=Relation Prediction, emb=word embeddings, cr=classification rules, sm=statistical models, nn=neural networks.)

Paper	ID	CC	RP	emb	cr	sm	nn	Annotation
MM2011	*	*	*		*	*		**Component** Premise/Conclusion/Non-argumentative **Relation** Support/Against/Conclusion/Other/None
PSI2020	*	*	*	*			*	**Component** Premise/Conclusion/Non-argument
NPC2017		*	*	*		*	*	**Component** Fact/Testimony/Value/Policy/Reference
GLT2021		*	*	*			*	**Relation** Reason/Evidence
GACS2015		*				*		**Component**
WPDL2019		*				*		reasoning roles (e.g., Evidence, Finding)
WSWA2019		*			*	*		**Relation**
WSW2020		*					*	logical connectives (e.g., positive/negative)
XSA2020	*	*		*		*	*	**Component**
XSA2021a		*		*			*	Issue/Reason/Conclusion/Non-IRC
XSA2021b		*		*			*	

4.1 Argument Information Detection

Although arguments are considered as their major proportion, legal documents (e.g., case-laws) still have redundant parts without argument information. The first task in an AM system is to shrink the scope of argumentative content as well as filter out unrelated parts.

MM2011 considered this to be binary classification: whether a proposition (segmented clause) is argumentative or not. A number of statistical machine learning (ML) classifiers were used: Naïve Bayes (NB), Maximum Entropy (ME), and Support Vector Machine (SVM) with n-grams, Parts Of Speech (POS) tagging, hand-crafted features, etc. They received best results (accuracy = 0.80) through the ME model and the NB classifier. Likewise, PSI2020 began similarly, but using the transformer-based neural network RoBERTa rather than traditional ML models. They adapted the pre-trained network for contextual word embedding features. To understand the performance of ML techniques on the CanLII corpus, XSA2020 designed an experiment to classify IRC labelled sentences and non-IRC sentences using Random Forest (RF), Long Short-Term Memory (LSTM), Convolutional Neural Network (CNN) and FastText. They achieved best weighted F1 of 0.72 on summaries and 0.94 on full case texts.

4.2   Argument Component Classification

This refers to the classification of segmented sentences (clauses) as particular types of argument components. In some works, the input texts have previously been identified as argumentative, or already filtered during preprocessing. In other cases, this is merged with argument information detection by adding an extra label (i.e., "N/A") and formulating it as a multi-classification problem.

MM2011 handled this task as a second text classification problem following argument information detection. The best results (premise F1 = 0.68, conclusion F1 = 0.74) were achieved by using Context-Free Grammar (CFG) and statistical classifiers (ME classifier and SVM). To simplify their experiment process, PSI2020 presumed all argumentative clauses had previously been successfully detected. Considering that a clause may act as a premise in one argument and conclusion in another, they divided this task into two binary classifications. They then applied separate RoBERTa models with the F1 measure reported individually (premise F1= 0.86, conclusion F1= 0.63).

The LLT Lab have built a variety of AM systems on the V/IP and BVA datasets. Using AM as a base module, GACS2015 introduced a legal document retrieval architecture where ten cases from the annotated V/IP corpus were used to train a classifier to predict component annotations of all non-gold-standard documents. This used NB, Decision Tree (DT), and Logistic Regression (LR) models with TF-IDF feature-vectors of n-grams, sub-sentences, etc. Their LR model reached the best Micro F1 (0.24) and Macro F1 (0.31), and DT achieved the best accuracy (0.97). In the study of the BVA corpus, WPDL2019 used a qualitative methodology to analyse a small sample (530 sentences) and developed rule-based scripts for component classification. They compared the result with other ML algorithms (NB, LR, and SVM) trained and tested on a large dataset (5,800 sentences). Both LR and SVM reached an average accuracy of 0.86. WSWA2019 presented an explainable classifier using Boolean search rules to categorise segmented legal text as argument components. They developed an interactive environment to create Boolean rules for both annotation and classification. One motivation for using rule-based classifiers was that they are more explainable than ML models, and required less labelled data. They trained four benchmark ML models (RF, SVM, FastText, and SKOPE-rules), which performed better than human-generated rules. WSW2020 also studied component classification on the BVA corpus from a linguistic polarity perspective. They designed a five-layer neural network with two evaluation datasets, a train-test cross validation on 50 decisions and a test-only experiment on 25 decisions. In two experiments, they achieved accuracy of 0.89 and 0.88 respectively.

Among the research of argument component (IRC-triple) classification on the CanLII corpus, XSA2020 measured three types of techniques: traditional ML model (RF), deep neural networks (LSTM, CNN), and FastText with GloVe embeddings. Among all the models, CNN and RF achieved the highest scores on case summaries (weighted F1 = 0.63) and full text (weighted F1 = 0.91). XSA2021a continued the exploration of deep neural networks. They used LSTM, CNN, BERT, and model combinations. Instead of manually mapping the IRC

annotations from case summaries to full texts, they investigated whether this process can be automatic. XSA2021b expanded the previous study, demonstrating that domain-specific pre-training corpora enhance BERT models' performance in Legal AM. They then merged BERT embeddings with a bidirectional LSTM (BiLSTM) network, and proved the position information enhancement on argument component classification. Although, compared to XSA2021a, the test scores decreased, XSA2021b suggested it was caused by lack of training data.

4.3   Argument Relation Prediction

Predicting argument relations is the most difficult part of the AM pipeline, aiming to discover relations between arguments and argument components. Since the argument relation annotations vary between corpora (see Section 3), in this task, we include the predictions of both inner relations which link between argument components and outer relations which link between arguments.

The final stage in MM2011 is to detect relations between full arguments, which requires the determination of the limits of individual arguments and relations with surrounding arguments. They studied argumentative parsing using rhetorical structure theory and POS tagging, then parsed the text by manually derived rules into their self-defined CFG. By parsing via this CFG, MM2011 reached an accuracy around 0.60 in detecting the complete argumentation structure. In contrast to MM2011, relation prediction in PSI2020 aimed to group argumentative clauses (components) into arguments where they are implicitly connected by relations. PSI2020 simplified this task as a sentence-pair classification problem to predict whether a pair of argumentative clauses belong to the same argument. This allows individual clauses to be recognised in multiple arguments. PSI2020 used a sliding window (size $= 5$) to generate the sentence pairs, and assumed that all the argumentative clauses have been identified successfully. The RoBERTa classifier reached an F1 of 0.51. PSI2020 explained an extra operation was still needed to arrange the identified pairs into arguments.

Since over 20% of the argument relations in CDCP do not suit the tree structure, NPC2017 transformed the pipeline into a document-level joint learning model, and represented the argumentation as factor graphs. They aimed to predict argument component types for sentences, and argument relations for sentence pairs. Several techniques (e.g., pre-defined rules, patterns in valid graph, etc.) were applied to constrain and train the model. To represent argument components and relations in the factor graph, various types of features (e.g., hand-crafted, contextual, lexical, etc.) were stored as variables. Using GloVe embeddings, NPC2017 built a linear structured SVM, and a BiLSTM network. The linear-SVM achieved the best results on component classification (F1 $= 0.73$) and relation prediction (F1 $= 0.27$). Inspired by NPC2017, GLT2021 designed a neural network with stacked modules, which jointly performed both component classification and relation prediction (also using GloVe embeddings). The neural network consisted of a residual network model, with an LSTM network, and an attention block. They tested a new prediction strategy, using multiple models

ensemble voting. In this case, they improved component classification F1 score to 0.79 and relation prediction F1 score to 0.30.

5   Conclusion and Outlook

In reviewing the development of Legal AM in the past decade, our work presents a comprehensive survey from two aspects: annotated legal corpora, and practical AM implementations. As well as identifying and analysing the available annotated corpora, our work also reviews the performance of previous ML techniques on Legal AM. During our study, we detected several remaining challenges and prospects which require future work, as follows.

Many previous Legal AM studies relied on rule-based or statistical models. Although researchers have begun switching to neural networks, much remains to be explored, especially when applying advanced NLP approaches. Supervised learning used by neural networks requires substantial annotated data, and the balance between system performance and the expert labour required for annotation is always an issue. Pre-trained NLP models (e.g., BERT) have shown strong performance on downstream tasks with limited corpora, which is a promising approach for Legal AM [36, 37].

Many annotation schemes are designed according to semantic rule and knowledge graph. At present, tools to visualise the retrieved argumentation details are still required. There is potential for NLP models and knowledge graphs to be merged together to enhance Legal AM and to present text information in a way that suits legal professionals better.

The pipeline structure remains the dominant design for Legal AM. Nevertheless, error propagation remains an unavoidable issue between tasks, whereby errors in earlier stages of the pipeline have a cascading effect on later stages. This is challenging for evaluation and for practical use. We suggest that other innovative methods and tools, like dependency parsing, multi-task learning, and graph neural networks, may replace the pipeline structure. These techniques have already achieved breakthroughs in general AM research [35, 7].

References

1. Besnard, P., Hunter, A.: A logic-based theory of deductive arguments. Artificial Intelligence **128**(1-2), 203–235 (2001)
2. Bibal, A., Lognoul, M., De Streel, A., Frénay, B.: Legal requirements on explainability in machine learning. Artificial Intelligence and Law **29**(2), 149–169 (2021)
3. Cabrio, E., Villata, S.: Five years of argument mining: a data-driven analysis. In: IJCAI. vol. 18, pp. 5427–5433 (2018)
4. Chalkidis, I., Androutsopoulos, I., Aletras, N.: Neural legal judgment prediction in english. In: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. pp. 4317–4323 (2019)
5. Cohen, J.: A coefficient of agreement for nominal scales. Educational and psychological measurement **20**(1), 37–46 (1960)

6. Dung, P.M.: On the acceptability of arguments and its fundamental role in non-monotonic reasoning, logic programming and n-person games. Artificial intelligence **77**(2), 321–357 (1995)
7. Galassi, A., Lippi, M., Torroni, P.: Multi-task attentive residual networks for argument mining. arXiv preprint arXiv:2102.12227 (2021)
8. Grabmair, M., Ashley, K.D., Chen, R., Sureshkumar, P., Wang, C., Nyberg, E., Walker, V.R.: Introducing luima: an experiment in legal conceptual retrieval of vaccine injury decisions using a uima type system and tools. In: Proceedings of the 15th international conference on artificial intelligence and law. pp. 69–78 (2015)
9. Lippi, M., Torroni, P.: Argument mining: A machine learning perspective. In: International Workshop on Theory and Applications of Formal Argumentation. pp. 163–176. Springer (2015)
10. Lippi, M., Torroni, P.: Argumentation mining: State of the art and emerging trends. ACM Transactions on Internet Technology (TOIT) **16**(2), 1–25 (2016)
11. Mochales, 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 12th international conference on artificial intelligence and law. pp. 21–30 (2009)
12. Mochales, R., Moens, M.F.: Study on the structure of argumentation in case law. In: Legal Knowledge and Information Systems, pp. 11–20. IOS Press (2008)
13. Mochales, R., Moens, M.F.: Argumentation mining. Artificial Intelligence and Law **19**(1), 1–22 (2011)
14. Nallapati, R., Manning, C.D.: Legal docket-entry classification: Where machine learning stumbles. In: 2008 Conference on Empirical Methods in Natural Language Processing. p. 438 (2008)
15. Niculae, V., Park, J., Cardie, C.: Argument mining with structured svms and rnns. In: Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). pp. 985–995 (2017)
16. Park, J., Blake, C., Cardie, C.: Toward machine-assisted participation in erulemaking: An argumentation model of evaluability. In: Proceedings of the 15th International Conference on Artificial Intelligence and Law. pp. 206–210 (2015)
17. Park, J., Cardie, C.: A corpus of erulemaking user comments for measuring evaluability of arguments. In: Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018) (2018)
18. Poudyal, P., Šavelka, J., Ieven, A., Moens, M.F., Gonçalves, T., Quaresma, P.: Echr: legal corpus for argument mining. In: Proceedings of the 7th Workshop on Argument Mining. pp. 67–75 (2020)
19. Savelka, J., Walker, V.R., Grabmair, M., Ashley, K.D.: Sentence boundary detection in adjudicatory decisions in the united states. Traitement automatique des langues **58**, 21 (2017)
20. Stab, C., Gurevych, I.: Annotating argument components and relations in persuasive essays. In: Proceedings of COLING 2014, the 25th international conference on computational linguistics: Technical papers. pp. 1501–1510 (2014)
21. Teruel, M., Cardellino, C., Cardellino, F., Alemany, L.A., Villata, S.: Legal text processing within the mirel project. In: 1st Workshop on Language Resources and Technologies for the Legal Knowledge Graph. p. 42 (2018)
22. Toulmin, S.E.: The uses of argument. Cambridge university press (2003)
23. Urchs, S., Mitrovic, J., Granitzer, M.: Design and implementation of german legal decision corpora. In: ICAART (2). pp. 515–521 (2021)

24. Walker, V., Foerster, D., Ponce, J.M., Rosen, M.: Evidence types, credibility factors, and patterns or soft rules for weighing conflicting evidence: Argument mining in the context of legal rules governing evidence assessment. In: Proceedings of the 5th Workshop on Argument Mining. pp. 68–78 (2018)
25. Walker, V.R.: A default-logic paradigm for legal fact-finding. Jurimetrics **47**, 193 (2006)
26. Walker, V.R., Carie, N., DeWitt, C.C., Lesh, E.: A framework for the extraction and modeling of fact-finding reasoning from legal decisions: lessons from the vaccine/injury project corpus. Artificial Intelligence and Law **19**(4), 291–331 (2011)
27. Walker, V.R., Han, J.H., Ni, X., Yoseda, K.: Semantic types for computational legal reasoning: propositional connectives and sentence roles in the veterans' claims dataset. In: Proceedings of the 16th edition of the International Conference on Articial Intelligence and Law. pp. 217–226 (2017)
28. Walker, V.R., Pillaipakkamnatt, K., Davidson, A.M., Linares, M., Pesce, D.J.: Automatic classification of rhetorical roles for sentences: Comparing rule-based scripts with machine learning. In: ASAIL@ ICAIL (2019)
29. Walker, V.R., Strong, S.R., Walker, V.E.: Automating the classification of finding sentences for linguistic polarity. In: ASAIL@ JURIX (2020)
30. Walton, D.: Argumentation theory: A very short introduction. In: Argumentation in artificial intelligence, pp. 1–22. Springer (2009)
31. Westermann, H., Savelka, J., Walker, V.R., Ashley, K.D., Benyekhlef, K.: Computer-assisted creation of boolean search rules for text classification in the legal domain. In: JURIX. pp. 123–132 (2019)
32. Xu, H., Šavelka, J., Ashley, K.D.: Using argument mining for legal text summarization. In: Legal Knowledge and Information Systems, pp. 184–193. IOS Press (2020)
33. Xu, H., Savelka, J., Ashley, K.D.: Accounting for sentence position and legal domain sentence embedding in learning to classify case sentences. In: Legal Knowledge and Information Systems, pp. 33–42. IOS Press (2021)
34. Xu, H., Savelka, J., Ashley, K.D.: Toward summarizing case decisions via extracting argument issues, reasons, and conclusions. In: Proceedings of the eighteenth international conference on artificial intelligence and law. pp. 250–254 (2021)
35. Ye, Y., Teufel, S.: End-to-end argument mining as biaffine dependency parsing. In: Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume. pp. 669–678 (2021)
36. Zhang, G., Lillis, D., Nulty, P.: Can Domain Pre-training Help Interdisciplinary Researchers from Data Annotation Poverty? A Case Study of Legal Argument Mining with BERT-based Transformers. In: Proceedings of the Workshop on Natural Language Processing for Digital Humanities (NLP4DH). pp. 121–130. Association for Computational Linguistics (2021)
37. Zhang, G., Nulty, P., Lillis, D.: Enhancing Legal Argument Mining with Domain Pre-training and Neural Networks. Journal of Data Mining and Digital Humanities **NLP4DH** (2022). https://doi.org/10.46298/jdmdh.9147