Applying Argument Extraction to Improve Legal Information Retrieval

Kevin D. Ashley
University of Pittsburgh School of Law
Pittsburgh, Pennsylvania, USA 15260
ashley@pitt.edu

Abstract

Argument extraction techniques can likely improve legal information retrieval. Any effort to achieve that goal should take into account key features of legal reasoning such as the importance of legal rules and concepts, support and attack relations among claims, and citation of authoritative sources. Annotation types reflecting these key features will help identify the roles of textual elements in retrieved legal cases in order to better inform assessments of relevance for users' queries. As a result, legal argument models and argument schemes will likely play a central part in the text annotation type system.

1 Introduction

With improved prospects for automatically extracting arguments from text, we are investigating whether and how argument extraction can improve legal information retrieval (IR). An immediate question in that regard is the role that argument models and argument schemes will play in achieving this goal.

For some time, researchers in Artificial Intelligence and Law have developed argument models, formal and dialectical process models to describe arguments and their relations. They have also implemented these models in computer programs that construct legal arguments. Some of these models employ argument schemes to provide semantics and describe reasonable arguments. Each scheme corresponds to a typical domain-specific inference sanctioned by the argument, a kind of *prima facie* reason for believing the argument's conclusion. See (Prakken, 2005, p. 234).

By and large, however, these argument models and schemes and their computational implementations have not had much of a practical effect on legal practice. A primary reason for this is the well-known bottleneck in representing knowledge from the legal texts (e.g., statutes, regulations, and cases) that play such an important role in legal practice in a form so that the the computational implementations can reason with them.

Meanwhile, legal information retrieval systems have proven to be highly functional. They provide legal practitioners with convenient access to millions of legal texts without relying on argument models or schemes, relying instead on Bayesian statistical inference based on term frequency. Users of legal information systems can submit queries in the form of a natural language description of a desired fact pattern and retrieve numerous relevant cases.

Useful as they are, however, legal information retrieval systems do not provide all of the functionality that practitioners could employ. What IR system users often want "is not merely IR, but AR", that is, "argument retrieval: not merely sentences with highlighted terms, but arguments and argument-related information. For example, users want to know what legal or factual issues the court decided, what evidence it considered relevant, what outcomes it reached, and what reasons it gave." (Ashley and Walker, 2013a).

Recently, IBM announced its Debater project, an argument construction engine which, given a corpus of unstructured text like Wikipedia, can automatically construct a set of relevant pro/con arguments phrased in natural language. Built upon the foundation of IBM's Jeopardy-game-winning Watson question answering system, the advent of Debater raises some interesting related questions. A central hypothesis of the Watson project was to answer questions based on shallow syntactic knowledge and its implied semantics. This was preferred to formally represented deep semantic knowledge, the acquisition of which is difficult and expensive (Fan et al., 2012). If Debater is applied to legal domains (*See, e.g.,*(Beck, 2014)), one wonders to what extent the same will be true of Debater. In particular, to what extent will explicit argumentation models and their schemes for the legal domain be necessary or useful for the effort to extract legal arguments? And, can techniques in Debater be adapted to improve legal IR?

2 Related Work

The seminal work on extracting arguments and argument-related information from legal case decisions is (Mochales and Moens, 2011). Operationally, the authors defined an argument as "a set of propositions, all of which are premises except, at most, one, which is a conclusion. Any argument follows an argumentation scheme. . . ." Using machine learning based on manually classified sentences from the Araucaria corpus, including court reports, they achieved good performance on classifying sentences as propositions in arguments or not and classifying argumentative propositions as premises or conclusions. Given a limited set of documents, their manually-constructed rule-based argument grammar also generated argument tree structures (Mochales and Moens, 2011).

In identifying argumentative propositions, Mochales and Moens achieved accuracies of 73% and 80% on two corpora, employing domain-general features (including, e.g., each word, pairs of words, pairs and triples of successive words, parts of speech including adverbs, verbs, modal auxiliaries, punctuation, keywords indicating argumentation, parse tree depth and number of subclauses, and certain text statistics.) For classifying argumentative propositions as premises or conclusions, their features included the sentence's length and position in the document, tense and type of main verb, previous and successive sentences' categories, a preprocessing classification as argumentative or not, and the type of rhetorical patterns occurring in the sentence and surrounding sentences (i.e., Support, Against, Conclusion, Other or None). Additional features, more particular to the legal domain included whether the sentence referred to or defined a legal article, the presence of certain argumentative patterns (e.g. "see", "*mutatis mutandis*", "having reached this conclusion", "by a majority") and whether the agent of the sentence is the plaintiff, the defendant, the court or other (Mochales and Moens, 2011).

Factors, stereotypical fact patterns that strengthen or weaken a side's argument in a legal claim, have been identified in text automatically. Using a HYPO-style CBR program and an IR system relevance feedback module, the SPIRE program retrieved legal cases from a text corpus and highlighted passages relevant to bankruptcy law factors (Daniels and Rissland, 1997). The SMILE+IBP program learned to classify case summaries in terms of applicable trade secret law factors (Ashley and Brüninghaus, 2009), analyzed automatically classified squibs of new cases, predicted outcomes, and explained the predictions. (Wyner and Peters, 2010) presents a scheme for annotating 39 trade secret case texts with GATE in terms of finer grained components (i.e., factoroids) of a selection of factors.

Using an argument model to assist in representing cases for conceptual legal information retrieval was explored in (Dick and Hirst, 1991). More recently, other researchers have addressed automatic semantic processing of case decision texts for legal IR, achieving some success in automatically:

- assigning rhetorical roles to case sentences based on 200 manually annotated Indian decisions (Saravanan and Ravindran, 2010),
- categorizing legal cases by abstract Westlaw categories (e.g., bankruptcy, finance and banking) (Thompson, 2001) or general topics (e.g., exceptional services pension, retirement) (Gonçalves and Quaresma, 2005),
- extracting treatment history (e.g., "affirmed", "reversed in part") (Jackson et al., 2003),
- determining the role of a sentence in the legal case (e.g., as describing the applicable law or the facts) (Hachey and Grover, 2006),
- extracting offenses raised and legal principles applied from criminal cases to generate summaries (Uyttendaele et al., 1998),
- extracting case holdings (McCarty, 2007), and
- extracting argument schemes from the Araucaria corpus such as argument from example and argument from cause to effect (Feng and Hirst, 2011).

We aim to develop and evaluate an integrated approach using both semantic and pragmatic (contextual) information to retrieve arguments from legal texts in order to improve legal information retrieval. We are working with an underlying argumentation model and its schemes, the Default Logic Framework (DLF), and a corpus of U.S. Federal Claims Court cases (Walker et al., 2011; Walker et al., 2014; Ashley and Walker, 2013a). Like (Mochales and Moens, 2011) and (Sergeant, 2013), we plan to:

1. Train an annotator to automatically identify propositions in unseen legal case texts,

2. Distinguish argumentative from non-argumentative propositions and classify them as premises or conclusions,

3. Employ rule-based or machine learning models to construct argument trees from unseen cases based on a manually annotated training corpus, but also to

4. Use argument trees to improve legal information retrieval reflecting the uses of propositions in arguments.

Before sketching our approach for the legal domain, however, we note that IBM appears to have developed more domain independent techniques for identifying propositions in documents and classifying them as premises in its Debater system.[1]

On any topic, the Debater's task is to "detect relevant claims" and return its "top predictions for pro claims and con claims." On inputting the topic, "The sale of violent videogames to minors should be banned," for example, Debater:

(1) scanned 4 million Wikipedia articles,

(2) returned the 10 most relevant articles,

(3) scanned the 3000 sentences in those 10 articles,

(4) detected those sentences that contained "candidate claims",

(5) "identified borders of candidate claims",

(6) "assessed pro and con polarity of candidate claims",

(7) "constructed a demo speech with top claim predictions", and

(8) was then "ready to deliver!"

Figure 1 shows an argument diagram constructed manually from the video recording of Debater's oral output for the example topic.

3 Key Elements of Legal Argument

Debater's argument regarding banning violent video games is meaningful but compare it to the *legal* argument concerning a similar topic in Figure 2. The Court in Video Software Dealers Assoc. v. Schwarzenegger, 556 F. 3d 950 (9th Cir. 2009), addressed the issue of whether California (CA) Civil Code sections 1746-1746.5 (the "Act"), which restrict sale or rental of "violent video games" to minors, were unconstitutional under the 1st and 14th Amendments of the U.S. Constitution. The Court held the Act unconstitutional. As a presumptively invalid content-based restriction on speech, the Act is subject to strict scrutiny and the State has not demonstrated a compelling interest.

In particular, the Court held that CA had not demonstrated a compelling government interest that "the sale of violent video games to minors should be banned." Figure 2 shows excerpts from the portion of the opinion in which the Court justifies this conclusion. The nodes contain propositions from that portion and the arcs reflect the explicit or implied relations among those propositions based on a fair reading of the text.

The callout boxes in Figure 2 highlight some key features of legal argument illustrated in the Court's argument:

1. Legal rules and concepts govern a court's decision of an issue.

2. Standards of proof govern a court's assessment of evidence.

3. Claims have support / attack relations.

4. Authorities are cited (e.g., cases, statutes).

5. Attribution information signals or affects judgments about belief in an argument (e.g., "the State relies").

6. Candidate claims in a legal document have different plausibility.

[1] *See, e.g.,* `http://finance.yahoo.com/blogs/the-exchange/ibm-unveils-a-computer-than-can-argue-181228620.html`. A demo appears at the 45 minute mark: `http://io9.com/ibms-watson-can-now-debate-its-opponents-1571837847`.

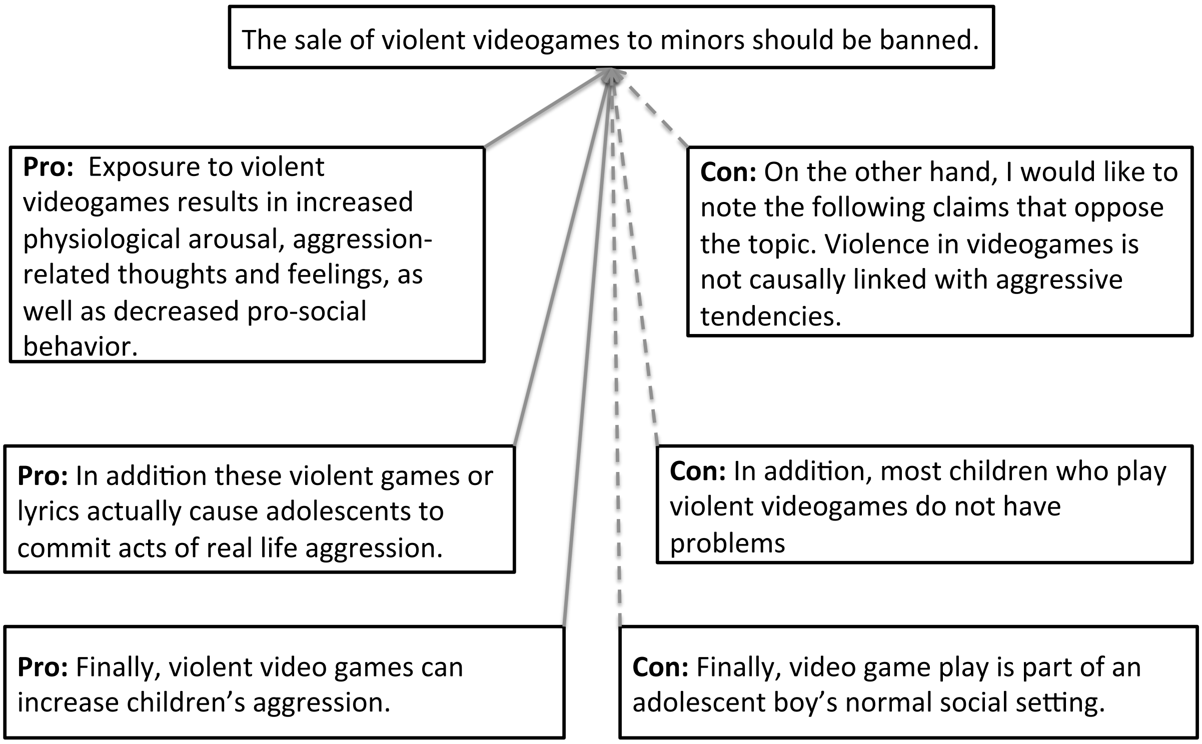

Figure 1: Argument Diagram of IBM Debater's Output for Violent Video Games Topic (root node)

Although the argument diagrams in Figures 1 and 2 address nearly the same topic and share similar propositions, the former obviously lacks these features that would be important in legal argument (and, as argued later, important in using extracted arguments to improve legal IR). Of course, on one level this is not surprising; the Debater argument is *not* and does not purport to be a legal argument.

On the other hand, given the possibility of applying Debater to legal applications and argumentation, it would seem essential that it be able to extract such key information. In that case, the question is the extent to which explicit argument models and argument schemes of legal reasoning would be useful in order to assist with the extraction of the concepts, relationships, and information enumerated above and illustrated in Figure 2.

4 Default-Logic Framework

Vern Walker's Default Logic Framework (DLF) is an argument model plus schemes for evidence-based legal arguments concerning compliance with legal rules. At the Research Laboratory for Law, Logic and Technology (LLT Lab) at Hofstra University, researchers have applied the DLF to model legal decisions by Court of Federal Claims "Special Masters" concerning whether claimants' compensation claims comply with the requirements of a federal statute establishing the National Vaccine Injury Compensation Program. Under the Act, a claimant may obtain compensation if and only if the vaccine caused the injury.

In order to establish causation under the rule of Althen v. Secr. of Health and Human Services, 418 F.3d 1274 (Fed.Cir. 2005), the petitioner must establish by a preponderance of the evidence that: (1) a "medical theory causally connects" the type of vaccine with the type of injury, (2) there was a "logical sequence of cause and effect" between the particular vaccination and the particular injury, and (3) a "proximate temporal relationship" existed between the vaccination and the injury. Walker's corpus comprises all decisions in a 2-year period applying the *Althen* test of causation-in-fact (35 decision texts, 15-40 pages per decision). In these cases, the Special Masters decide which evidence is relevant to which issues of fact, evaluate the plausibility of evidence in the legal record, organize evidence and draw reasonable inferences, and make findings of fact.

The DLF model of a single case "integrates numerous units of reasoning" each "consisting of one

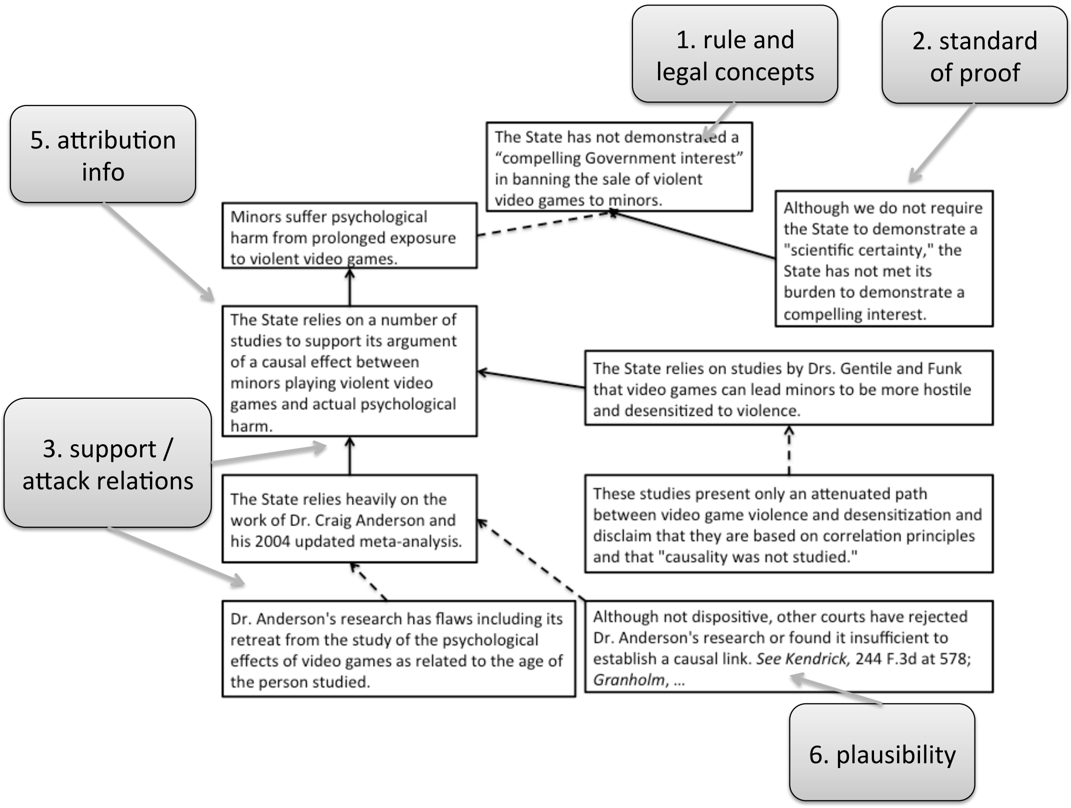

Figure 2: Diagram Representing Realistic Legal Argument Involving Violent Video Games Topic

conclusion and one or more immediately supporting reasons (premises)" and employing four types of connectives (min (and), max (or), evidence factors, and rebut) (Walker et al., 2014). For example, Figure 3 shows an argument diagram representing the excerpt of the the DLF model of the special master's finding in the case of Cusati v. Secretary of Health and Human Services, No. 99-0492V (Office of Special Masters, United States Court of Federal Claims, September 22, 2005) concerning whether the first *Althen* condition for showing causation-in-fact is satisfied.

The main point is that the DLF model of a legal argument and its argument schemes represent the above-enumerated key features of legal argument. As illustrated in the callout boxes of Figure 3, the model indicates: (1) the 1st *Althen* rule and causation-in-fact concept that govern the decision of the causation issue, (2) the preponderance of evidence standard of proof governing the court's assessment, (3) support relations among the propositions, the Special Master having recorded no counterarguments, (4) citation to the statute, 42 USC 300aa-11(c)(1)(C)(ii)), and to the *Althen* and *Shyface* case authorities, (5) some attribution information that signals judgments about the Special Master's belief in an argument (e.g., "Dr. Kinsbourne and Dr. Kohrman agree"), and (6) four factors that increase plausibility of the claim of causation.

5 Legal Argument and Legal IR

Legal decisions contain propositions and arguments how to "prove" them. Prior cases provide examples of how to make particular arguments in support of similar hypotheses and of kinds of arguments that have succeeded, or failed, in the past. Consider a simple query discussed in (Ashley and Walker, 2013a): Q1: "MMR vaccine can cause intractable seizure disorder and death."

An attorney/user in a new case where an injury followed an MMR vaccination might employ this query to search for cases where such propositions had been addressed. Relevant cases would add confidence that the propositions and accompany-

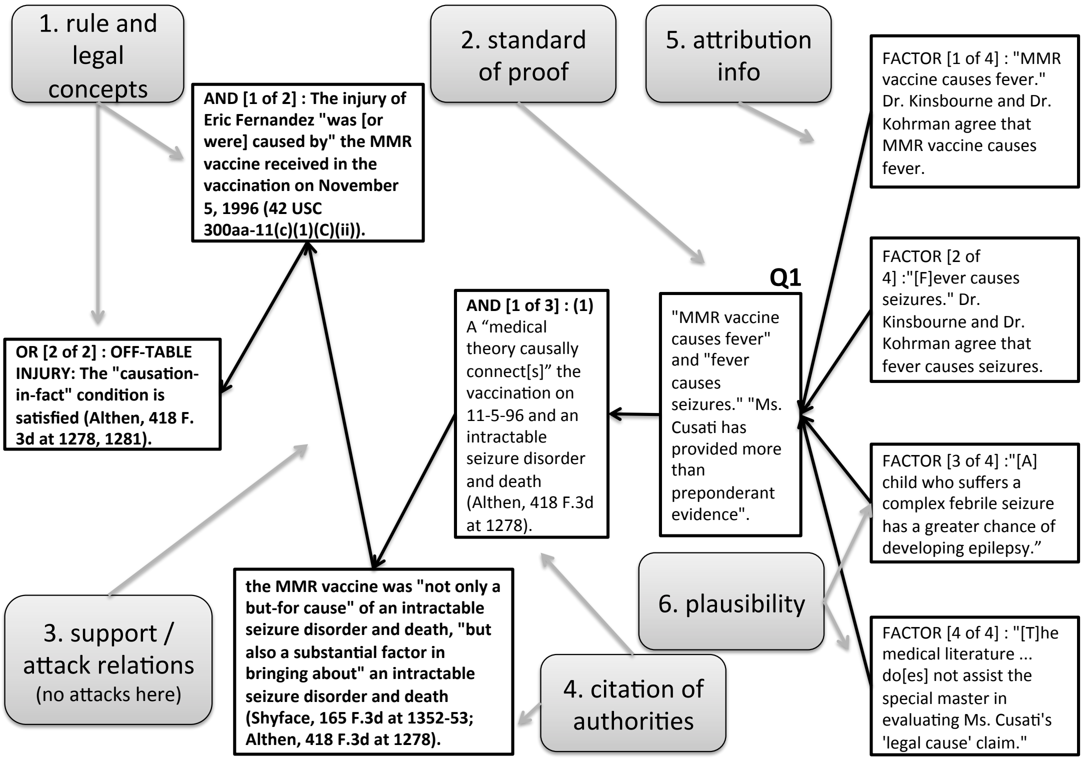

Figure 3: Diagram of DLF Model of Special Master's Finding in *Cusati* Case re 1st *Althen* Condition

ing arguments were reasonable and had been successful.

Importantly, the cases retrieved will be more relevant to the extent that the proposition is used in a similar argument. That is, they will be more relevant to the extent that the proposition plays roles in the case arguments similar to the role in which the attorney intends to use it in an argument about the current case.

An argument diagram like that of Figure 3 can illustrate the effect of the six key elements of legal reasoning illustrated above on how relevant a retrieved case is to a user's query. The diagram shows a legal argument in which the proposition corresponding to Q1 plays a role in the *Cusati* case as an evidence-based finding of the Special Master, namely, that "MMR vaccine causes fever" and "fever causes seizures."

Such diagrams have a "legal rule-oriented" direction (i.e., to the left in Figure 3) and an "evidentiary factors-oriented" direction (i.e., to the right in this diagram). For instance, an attorney whose client sustained seizures after receiving the MMR vaccine probably knows that he/she will have to satisfy a requirement of causation. The attorney may not know, however, what legal standard defines the relevant concept of causation or what legal authority may be cited as an authoritative source of the standard. In that situation, retrieved cases will likely be more relevant to the extent that that they fill in the legal rule-oriented direction, relative to a proposition similar to the one marked "Q1", with *legal rules* about the *concept* of causation and *citations* to their *authoritative sources*.

If the attorney is unsure of the kinds of evidence that an advocate should employ in convincing a Special Master to make the finding of fact on causation or of the relevant standard of proof for assessing that evidence of causation, retrieved cases will be more relevant to the extent that they fill in the evidentiary factors-oriented direction, relative to a proposition similar to the one marked "Q1", with evidentiary factors and an identification of the *standard of proof*.

The attorney may be interested in better understanding how to improve the *plausibility* of a proposition about causation as an evidence-based finding. Cases will be more relevant to the extent that they contain evidentiary factors that *support* such a finding. An attorney interested in attacking the *plausibility* of the evidence-based finding might be especially interested in seeing cases involving examples of evidentiary factors that *attack* such a finding.

Finally, the cases will be more relevant to the extent that the proposition similar to the one marked "Q1" concerning MMR vaccine's causing injury is *attributable* to the Special Master as opposed merely to some expert witness's statement.

6 Specifying/Determining Propositions' Argument Roles

The importance of a proposition's argument role in matching retrieved cases to users' queries raises two questions: (1) How does the user specify the target propositions and their argumentative roles in which he is interested? (2) How does a program determine the roles that propositions play in retrieved case arguments?

An argument diagram like that of Figure 3 may play a role in enabling users to specify the arguments and propositions in which they are interested. One can imagine a user's inputting a query by employing a more abstract version of such a diagram. For instance, in the Query Input Diagram of Figure 4, the nodes are labeled with, or refer to, argument roles. These roles include:

Legal Rule: sentences that state a legal rule in the abstract, without applying the rule to the particular case being litigated

Ruling/Holding: sentences that apply a legal rule to decide issues presented in the particular case being litigated

Evidence-Based Finding: sentences that report a trier-of-fact's ultimate findings regarding facts material to the particular case being litigated

Evidence-Based Reasoning: sentences that report the trier-of-fact's reasoning in assessing the relevant evidence and reaching findings regarding facts material to the particular case being litigated (e.g., evidentiary factors)

Evidence: sentences that describe any type of evidence legally produced in the particular case being litigated, as part of the proof intended to persuade the trier-of-fact of alleged facts material to the case (e.g., oral testimony of witnesses, including experts on technical matters; documents, public records, depositions; objects and photographs)

Citation: sentences that credit and refer to authoritative documents and sources (e.g., court decisions (cases), statutes, regulations, government documents, treaties, scholarly writing, evidentiary documents)

In the "text", "concept", and "citation" slots of the appropriate nodes of the query input diagram, Figure 4, users could specify the propositions, concepts, or citations that they know or assume and check the targeted nodes in the directions (rule-oriented or evidentiary-factors-oriented) or ranges that they hope to fill through searching for cases whose texts satisfy the diagram's argument-related constraints. In effect, the diagram will guide the IR system in ranking the retrieved cases for relevance and in highlighting their relevant parts.

Regarding the second question, concerning how a program will determine propositions' argument roles in case texts, that is the third task that Mochales and Moens addressed with a rule-based grammar applied to a small set of documents. While their rules employed some features particular to legal argument, (e.g., whether a sentence referred to a legal article) one imagines that additional features would be needed, pertaining to legal argument or to the regulated domain of interest. These features would become the predicates of additional grammar rules or be annotated in training cases for purposes of machine learning.

The legal argument roles listed above are a first cut at a more comprehensive enumeration of the types of legal argument features with which to annotate legal case texts in an Unstructured Information Management Architecture (UIMA) annotation pipeline for purposes of extracting argument information and improving legal IR.

UIMA, an open-source Apache framework, has been deployed in several large-scale government-sponsored and commercial text processing applications, most notably, IBM's Watson question answering system (Epstein et al., 2012). A UIMA

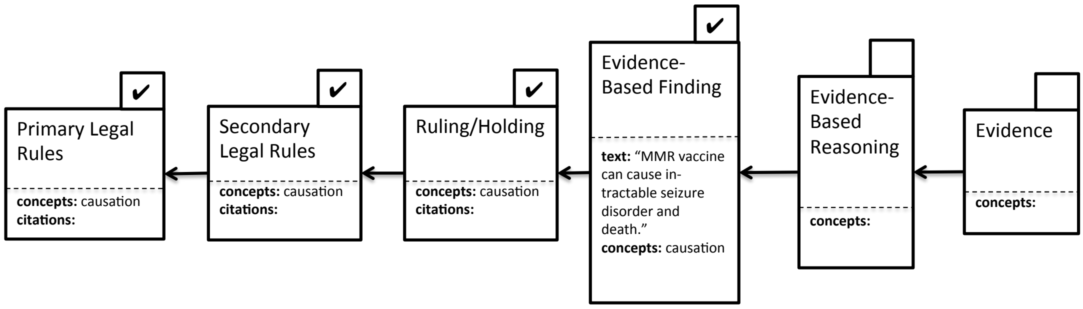

Figure 4: Sample Query Input Diagram

pipeline is an assemblage of integrated text annotators. The annotators are "a scalable set of cooperating software programs, ..., which assign semantics to some region of text" (Ferrucci, 2012), and "analyze text and produce annotations or assertions about the text" (Ferrucci et al., 2010, p. 74).

A coordinated type system serves as the basis of communication among these annotators; a type system embodies a formalization of the annotators' analysis input and output data (Epstein et al., 2012, p. 3). In (Ashley and Walker, 2013b) and (Ashley and Walker, 2013a) the authors elaborate three additional bases for annotations, which, with further refinement, may serve as a conceptual substrate for the annotation types listed above:

1. DLF annotations, as suggested in Figure 3, capture "(i) the applicable statutory and regulatory requirements as a tree of authoritative rule conditions (i.e., a "rule tree") and (ii) the chains of reasoning in the legal decision that connect evidentiary assertions to the special master's findings of fact on those rule conditions (Walker et al., 2011)."

2. Annotations in terms of presuppositional information that "identifies entities (e.g., types of vaccines or injuries), events (e.g., date of vaccination or onset of symptoms) and relations among them used in vaccine decisions to state testimony about causation, assessments of probative value, and findings of fact." (Ashley and Walker, 2013a).

3. Annotations of of argument patterns based on: inference type (e.g., deductive or statistical), evidence type (e.g., legal precedent, policy, fact testimony), or type of weighing of source credibility to resolve evidentiary discrepancies (e.g., in terms of expert vs. expert or of adequacy of explanation) (Walker et al., 2014) .

If we succeed in designing a system of coordinated legal annotation types and operationalizing a UIMA annotation pipeline, we envision adding a module to a full-text legal IR system. At *retrieval time* it would extract semantic / pragmatic legal information from the top *n* cases returned by a traditional IR search and re-rank returned cases to reflect the user's diagrammatically specified argument need. The module would also summarize highly ranked cases and highlight argument-related information (Ashley and Walker, 2013a). Since the module processes the texts of cases returned by the information retrieval system, no special knowledge representation of the cases in the IR system database is required; the knowledge representation bottleneck will have been circumvented.

7 Conclusion

According to Wittgenstein, meaning lies in the way knowledge is used. Legal argument models and argument schemes can specify roles for legal propositions to play (and, interestingly, Stephen Toulmin was a student of Wittgenstein.) Thus, researchers can enable machines to search for and use legal knowledge intelligently in order, among other things, to improve legal information retrieval.

Although IBM Debater may identify argument propositions (e.g., claims), legal argument schemes could help it to address legal rules and concepts, standards of proof, internal support and attack relations, citation of statutory and case authorities, attribution, and plausibility. Open questions include the extent to which legal expert knowledge will be needed in order to operationalize argument schemes to extract arguments from legal case texts.

Acknowledgments

My colleagues Vern Walker, Matthias Grabmair, and Eric Nyberg make this work possible.

References

K. Ashley and S. Brüninghaus. 2009. Automatically classifying case texts and predicting outcomes. *Artificial Intelligence and Law*, pages 125–165.

K. Ashley and V. Walker. 2013a. From information retrieval (IR) to argument retrieval (AR) for legal cases: Report on a baseline study. In K. Ashley, editor, *JURIX*, volume 259 of *Frontiers in Artificial Intelligence and Applications*, pages 29–38. IOS Press.

K. Ashley and V. Walker. 2013b. Toward constructing evidence-based legal arguments using legal decision documents and machine learning. In *Proc. 14th Int'l Conf. on Artificial Intelligence and Law*, ICAIL '13, pages 176–180, New York, NY, USA. ACM.

S. Beck. 2014. Emerging technology shapes future of law. `http://www.americanlawyer.com/id=1202664266769/Emerging-Technology-Shapes-Future-of-Law`. Accessed: 2014-09-20.

J. Daniels and E. Rissland. 1997. Finding legally relevant passages in case opinions. In *ICAIL*, pages 39–46.

J. Dick and G. Hirst. 1991. A case-based representation of legal text for conceptual retrieval. In *Proceedings, Workshop on Language and Information Processing*, American Society for Information Science, pages 93–102.

EA Epstein, MI Schor, BS Iyer, A. Lally, EW Brown, and J. Cwiklik. 2012. Making Watson fast. *IBM J. Res. and Dev.*, 56(3.4):15–1.

J. Fan, A. Kalyanpur, DC Gondek, and DA Ferrucci. 2012. Automatic knowledge extraction from documents. *IBM J. Res. and Dev.*, 56(3.4):5–1.

V. Feng and G. Hirst. 2011. Classifying arguments by scheme. In Dekang Lin, Yuji Matsumoto, and Rada Mihalcea, editors, *ACL*, pages 987–996. The Association for Computer Linguistics.

D. Ferrucci, E. Brown, J. Chu-Carroll, J. Fan, D. Gondek, A. Kalyanpur, A. Lally, J. W. Murdock, E. Nyberg, J. Prager, N. Schlaefer, and C. Welty. 2010. Building Watson: An overview of the DeepQA project. *AI Magazine*, 31(3):59–79.

D. Ferrucci. 2012. Introduction to "This is Watson". *IBM J. Res. and Dev.*, 56(3.4):1–1.

T. Gonçalves and P. Quaresma. 2005. Is linguistic information relevant for the classification of legal texts? In *Proc. 10th Int'l Conf. on AI and Law*, ICAIL '05, pages 168–176, NY, NY. ACM.

B. Hachey and C. Grover. 2006. Extractive summarisation of legal texts. *Artificial Intelligence and Law*, 14(4):305–345.

P. Jackson, K. Al-Kofahi, A. Tyrrell, and A. Vachher. 2003. Information extraction from case law and retrieval of prior cases. *Artificial Intelligence*, 150(1-2):239–290, November.

L.T. McCarty. 2007. Deep semantic interpretations of legal texts. In *Proc. 11th Int'l Conf. on AI and Law*, ICAIL '07, pages 217–224, NY, NY. ACM.

R. Mochales and M.-F. Moens. 2011. Argumentation mining. *Artificial Intelligence and Law*, 19(1):1–22.

H. Prakken. 2005. AI & Law, logic and argument schemes. *Argumentation*, 19(3):303–320.

M. Saravanan and B. Ravindran. 2010. Identification of rhetorical roles for segmentation and summarization of a legal judgment. *Artificial Intelligence and Law*, 18(1):45–76.

A. Sergeant. 2013. Automatic argumentation extraction. In et al. P. Cimiano, editor, *ESWC*, volume 7882 of *Lecture Notes in Computer Science*, pages 656–660. Springer.

P. Thompson. 2001. Automatic categorization of case law. In *Proc. 8th Int'l Conf. on AI and Law*, ICAIL '01, pages 70–77, NY, NY. ACM.

C. Uyttendaele, M.-F. Moens, and J. Dumortier. 1998. Salomon: Automatic abstracting of legal cases for effective access to court decisions. *Artificial Intelligence and Law*, 6(1):59–79.

V. Walker, N. Carie, C. DeWitt, and E. Lesh. 2011. 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*, pages 291–331.

V. Walker, K. Vazirova, and C. Sanford. 2014. Annotating patterns of reasoning about medical theories of causation in vaccine cases: Toward a type system for arguments. In *Proc. 1st Workshop on Argumentation Mining, ACL 2014*.

A. Wyner and W. Peters. 2010. Lexical semantics and expert legal knowledge towards the identification of legal case factors. In *Proc. 23d Conf. on Legal Knowledge and Information Systems: JURIX 2010*, pages 127–136, Amsterdam. IOS Press.