Toward Constructing Evidence-Based Legal Arguments Using Legal Decision Documents and Machine Learning

Kevin D. Ashley
University of Pittsburgh
Intelligent Systems Program
Pittsburgh, PA, USA
001-412-624-7496
ashley@pitt.edu

Vern R. Walker
Maurice A. Deane School of Law
Hofstra University
Hempstead, NY, USA
001-516-463-5165
vern.r.walker@hofstra.edu

ABSTRACT

This paper explores how to extract argumentation-relevant information automatically from a corpus of legal decision documents, and how to build new arguments using that information. For decision texts, we use the Vaccine/Injury Project (V/IP) Corpus, which contains default-logic annotations of argument structure. We supplement this with presuppositional annotations about entities, events, and relations that play important roles in argumentation, and about the level of confidence that arguments would be successful. We then propose how to integrate these semantic-pragmatic annotations with syntactic and domain-general semantic annotations, such as those generated in the DeepQA architecture, and outline how to apply machine learning and scoring techniques similar to those used in the IBM Watson system for playing the Jeopardy! question-answer game. We replace this game-playing goal, however, with the goal of learning to construct legal arguments.

Keywords
Legal argumentation; Text annotation; Default-logic framework; Presuppositional annotation; DeepQA; IBM Watson.

1. INTRODUCTION AND OVERVIEW

Typically, a set of legal rules establishes standards of compliance, using domain-specific and open-textured concepts – i.e., concepts that are not defined in terms of rules, and whose application requires interpretation. Often, the interpretation of the concepts’ meaning evolves through a process that authoritatively applies the rules to new situations. Participants use previously decided cases to argue whether new situations satisfy the rule conditions.

Such a decision process produces a corpus of decision texts (legal decision documents) that are relevant to argumentation in new cases, by providing: authoritative applications of the rule conditions and concepts to identified situations; a ground truth for testing predictions about outcomes in new cases with new evidence; patterns for successful and unsuccessful argumentation; and guidance in retrieving, extracting, and organizing evidence for new arguments and new situations. The usual alternative to using decision texts to obtain this information is interviewing legal domain experts. Capturing their knowledge, however, may be less authoritative, more time-consuming and more expensive than working with decision texts. Moreover, decision texts can provide syntactic and semantic variability for machine learning.

The strategic hypothesis underlying this paper is that the extraction of argumentation-relevant information from decision texts, and the use of such information to construct evidence-based legal arguments, can be automated using natural language processing tools, such as those involved in the DeepQA architecture of the IBM Watson system. DeepQA uses refined techniques for machine learning from unstructured sources annotated in syntactic and domain-general semantic terms, and the IBM Watson system has used this architecture to successfully play Jeopardy! against human experts. [1; 2; 3; 7]. Jeopardy! is a popular game in which the player must respond to a clue (stated in the form of an answer) by supplying an answer in the form of a question (the question to which the clue is the correct answer). We propose using the DeepQA architecture to generate arguments about findings of fact based on factual evidence (evidence-based legal arguments), as contrasted with arguments about rules based on principles or policies.

We propose combining two complementary approaches to legal-domain argumentation analysis. First, default-logic models can be used to represent (i) the semantics of statutory and regulatory requirements as trees of rule conditions (i.e., a “rule tree” containing inferences based on authority) and (ii) the chains of reasoning in legal decisions that connect the evidentiary input to the findings of fact (i.e., inferences based on common sense, scientific method, etc.). This approach decomposes a proposition of interest (e.g., that a statutory provision has been satisfied) into an inference tree consisting of legal propositions and supporting factual assertions. The rule tree increases the granularity of the legal concepts to the point where they can be linked closely to common sense reasoning about facts, and thereby to more effective and efficient search, extraction, and reporting. For example, although high-level legal concepts (e.g., “entitled to compensation”) would pose a vague retrieval task, lower-level concepts into which they are decomposed should be more tractable (e.g., “medical condition,” “time of onset”).

A second, complementary approach to argumentation analysis is to annotate two kinds of presuppositional information that help identify background assumptions taken for granted in making evidentiary assertions. The first is information relevant to argumentation function – such as the entities (e.g., types of vaccines or injuries), events (e.g., date of vaccination or onset of symptoms) and relations among them that play pivotal roles in stating evidence relevant to legal compliance. The second kind of
presuppositional information is relevant to the level of confidence that an argument about compliance or non-compliance with legal requirements would be successful (e.g., a pattern of argument that has been successful or unsuccessful in prior cases).

Combining the default-logic models of decision documents and the presuppositional information could help design scoring functions of the type that DeepQA employs for scoring candidate answers. These would score assertions extracted from decisional and non-decisional documents, from the perspective of using them to construct legal arguments about new problems. In addition, the identification of argumentation in legal texts and the construction of new arguments for new situations require the study of language use in context (pragmatics) [5]. Presuppositions about the legal rule system and about the use of sentences to make findings of fact, for example, as well as patterns of successful argument used in the past within a particular legal context, play important roles in understanding and constructing legal argumentation.

2. THE DECISION-DOCUMENT CORPUS

Decision documents from the Vaccine/Injury Project (V/IP) Corpus report legal determinations about whether claims for compensation comply with legal requirements of the National Vaccine Injury Compensation Program (VICP).1 A critical legal issue is causation: a petitioner filing a claim should receive compensation if, but only if, the vaccine received did in fact cause the injury. If the government contests a claim, “special masters” attached to the Court of Federal Claims decide which evidence is relevant to which issues of fact, evaluate the plausibility of the evidence in the legal record, organize that evidence and draw reasonable inferences, and make findings of fact. In 2005, the case of Althen v. Secretary of Health and Human Services (reported at 418 F.3d 1274 (Fed.Cir. 2005)) codified three substantive conditions (sub-issues to be proved) for drawing an inference of causation in the most complex vaccine cases. The petitioner must establish, by a preponderance of the evidence: (1) that a “medical theory causally connects” the type of vaccine with the type of injury; (2) that there was a “logical sequence of cause and effect” between the particular vaccination and the particular injury; and (3) that a “proximate temporal relationship” existed between the vaccination and the injury (see Althen at 1278). The V/IP Corpus comprises every decision filed during a 2-year period, in which the special master applied the 3-prong test of causation-in-fact enunciated by Althen (a total of 35 decision texts, typically 15 to 40 pages per decision).

3. DLF ANNOTATIONS

The Default-Logic Framework (DLF) annotations included in the V/IP Corpus represent the propositional inference structure that is found within a decision document, including the legal rules that govern the case and the fact-finding in the case. Each case’s logic model (or “case model”) is divided into two major parts [6]: the governing system of legal rules and the factfinder’s assessment of the evidence in the case. The governing system of legal rules is formally represented as an inverted “rule tree” (a directed acyclic graph with the root node at the top). The top node of the tree is the ultimate issue to be proved by the petitioner—namely, that “the petitioner is entitled to compensation under the National Vaccine Injury Compensation Program.” Nodes in the rule tree represent the propositional conditions established by the

In modeling the reported reasoning in any particular vaccine decision, the LLT Lab extracts from the decision document all sentences that express what the special master considered significant in reaching findings on the three causation conditions. Such sentences may express statements made by testifying witnesses, contained in evidentiary documents, or made by the special master (e.g., describing a witness’s demeanor). In deciding a case, the special master explicitly or implicitly organizes these assertions, assigns them degrees of plausibility, and uses them to support her findings of fact.

The LLT Lab represents both evidence and findings as “evidentiary assertions.” A finding of fact is attached to the appropriate terminal rule condition, and the supporting evidentiary reasoning is attached to the finding (thus extending the rule tree to become a complete inference tree). The evidentiary assertions are represented as nodes in the “evidence assessment” portion of a branch, and edges are implications. Evidentiary assertions have plausibility-values on an ordinal, seven-valued scale, with values of “highly plausible” / “very plausible” / “slightly plausible” / “undecided” / “slightly implausible” / “very implausible” / “highly implausible.” Plausibility connectives, which correspond to the logical connectives, include: “MIN,” “MAX,” “REBUT,” and “EVIDENCE FACTORS.” The researchers assign plausibility-values to the terminal or leaf assertions in the inference tree, so that the plausibility-values and truth-values throughout the model reflect the factfinder’s evaluation. The evidence assessment inference trees for the 35 vaccine decisions in the V/IP Corpus contain over 1,000 assertion nodes, with the vast majority of those being attached to the three branches of the Althen causation conditions.

The process of extracting the DLF structure of a decision document can be decomposed into various tasks, including: (A)
the identification/annotation of text sentences that state findings of fact on critical rule conditions; (B) the identification/annotation of text sentences with respect to their “horizontal relevance” (i.e., the particular branch of the rule tree in which the sentences play an important inferential role); (C) within each branch of the rule tree, the identification/annotation of the horizontally relevant sentences with respect to their “vertical relevance” (i.e., their location in the inferential levels of support within the branch); and (D) the extraction from each of the sentences in steps (A)-(C) of the precise content or information (the evidentiary assertion) that plays an inferential role in the reasoning.

4. CASE EXAMPLE

As an example of a vaccine case and decision document, we use the decision of the special master in Casey v. Secretary of Health and Human Services (Office of Special Masters, Court of Federal Claims, No. 97-612V (December 12, 2005)). In Casey, the petitioner alleged that her varicella vaccination (containing a live, attenuated virus) caused encephalomyeloradiculoneuropathy, an autoimmune reaction affecting her central and peripheral nervous systems. She argued that the vaccine caused this adverse condition through alternative causal chains, which may have worked in combination: a direct viral infection and/or an immune-mediated inflammatory response. Special Master Sweeney decided all three of the Althen causation conditions in favor of the petitioner, and ultimately decided the entire case for the petitioner.

To illustrate DLF annotation, the Casey decision contains the following passage (at p. 27):

[1] In sum, petitioner was vaccinated against varicella on June 9, 1995. [2] The attenuated virus in the varicella vaccine both directly attacked petitioner’s nervous system and caused an immune-mediated inflammatory response in her nervous system. [3] As a result, within four weeks of her varicella vaccination, petitioner began to experience the onset of symptoms of her encephalomyeloneuritis.

The position of this passage in the document indicates that these sentences express findings of fact of the Special Master, and not merely the Special Master’s recitation of a witness’s testimony.

Figure 2 displays the logic model of the top-level evidence in the Casey decision concerning the second Althen condition. The rule condition for the branch is shown in bold-font at the top, with the branch finding of the Special Master directly under it. All other assertions in Figure 2 are horizontally relevant to this finding and contain only the inferentially relevant information from the English sentences. The vertical relevance is shown spatially, with the conclusion (parent node) at the top of an indented list of supporting assertions (children). This excerpt lists four assertions extracted from this passage, each preceded by “MIN,” which indicates that these four assertions are connected as siblings on the same level of the inference tree by the plausibility connective MIN. These four MIN assertions support the conclusion “MAX [1 of 2]: [T]he varicella vaccine caused … .” (The annotation “MAX [1 of 2]” indicates that there exists in the case model an alternative line of reasoning that parallels this one.)

Each assertion in Figure 2 contains additional annotations. For example, the annotated assertion, “[P]etitioner was vaccinated against varicella on June 9, 1995,” provides the information about: the plausibility-value assigned to this assertion (“Very Plausible”) as indicated by the color of the circle icon at the beginning of the assertion; three siblings connected with it to their parent by conjunction (as indicated by the “MIN [1 of 4]” annotation before the content of the assertion); the citation for the natural language text represented by this assertion (from which it was extracted) (i.e., page 27 of the decision as indicated by “C: 27” in the brackets following the assertion content); the source of the assertion, in the sense of the person who believes the assertion is very plausible (here the Special Master, as indicated by “S: finding of Special Master” in the brackets following the content of the assertion); the basis for the assertion (here an exhibit submitted by the petitioner, as indicated by “B: petitioner’s exhibit” in the brackets following the content of the assertion); and the plausibility-value assigned to this assertion as stipulated by the modeler (which here indicates that this is a terminal or leaf assertion, unsupported by any further reasoning in the model, as indicated by the line immediately below the assertion, stating “Very Plausible” and showing the appropriate colored icon).

5. PRESUPPOSITIONAL ANNOTATIONS

Table 1 provides a few examples of presuppositional information of interest in constructing evidence-based arguments in V/IP cases—examples of domain-specific semantic relations, both real world and linguistic, between typed objects and entities.

<table>
<thead>
<tr>
<th>Semantic Relations</th>
<th>Meaning (objects or event referents)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Covered-vaccine</td>
<td>a vaccine covered by the VICP</td>
</tr>
<tr>
<td>2. Specific-date</td>
<td>a specific month, day, year</td>
</tr>
<tr>
<td>3. Specific-vaccination</td>
<td>a vaccination on a Specific-date</td>
</tr>
<tr>
<td>4. Generic-injury</td>
<td>a type of injury, adverse condition or disease</td>
</tr>
</tbody>
</table>
Such presuppositional annotations could be used to score new documents for argumentation relevance. Other annotations need to provide contextual (pragmatic) information about, for example, the use of sentences to state legal findings of fact or to apply a legal policy in assessing evidence.

6. INTEGRATING LEGAL AND DEEPQA ANNOTATIONS

To support machine learning to construct evidence-based arguments, both types of legal annotation would be used to develop semantic-pragmatic analytics that receive input from a DeepQA system like Watson. DeepQA is an extensible architecture whose overarching principles include integrating shallow and deep knowledge, using a wide range of loosely coupled analytics, and pervasively estimating confidence in hypotheses about solutions [3:68]. Of particular importance in our project are the principal functions of four DeepQA components: (1) an English Slot Grammar (ESG) that provides syntactic-semantic input to (2) a predicate-argument structure (PAS) builder, which in turn provides input for (3) pattern-based relation extraction; these three components are used to create (4) a knowledge base (PRISMATIC) useful in playing the Jeopardy! question-answer game. In our project, however, this game-playing goal is replaced by the goal of constructing evidence-based legal arguments. In essence, the ESG parser and the PAS builder would provide input to relation extraction that is guided by the presuppositional and DLF annotations described in prior sections, to create a knowledge base useful in constructing legal arguments (e.g., a VACCINES knowledge base instead of PRISMATIC). This section briefly discusses the principal functions performed by these DeepQA components, and how the V/IP Corpus and annotations would adapt these functions for argument-building.

(1) Syntactic-semantic parsing. The DeepQA ESG parser uses a pipeline of tokenization and segmentation, morphological analysis, and syntactic analysis in order to produce a dependency tree [4-2]. The tree structure can include both shallow syntactic parses (e.g., the headword-modifier structure) and deep word-sense predications (identified using a lexicon of sense frames that define SG word senses) [4-2-4]. ESG uses mainly high-level semantic types [4-5]. The SG syntactic analysis converts the morphologically analyzed tokens into phrases, agglomerates phrases into multieord phrases, and assigns each phrase (intermediate or final) a parse score that numerically ranks the possible final parses [4-7].

(2) Predicate-argument abstraction. The PAS builder simplifies the ESG parse by removing inessential semantic distinctions [4-8]. One objective is to ensure that if multiple sentences with different ESG parses provide the same information (from the perspective of question-answering), then these sentences should have an identical PAS structure. If a relation-detection rule written over PAS (discussed below) matches any one of these sentences, it would match all of the others as well. The V/IP corpus provides not only relevant sentences extracted from decision documents, but also the evidentiary assertions extracted from those sentences. The assertions contain the core information used in the reasoning. For example, in sentence [3] from the case excerpt, the phrase “within four weeks of her varicella vaccination” is relevant to Althen condition 3, but not directly to Althen condition 2, and so is omitted from the associated assertion shown in Figure 2.

The V/IP corpus of decision texts, extracted findings of fact, and finding-supporting sentences would supply the information needed to construct a correspondence between the parsed sentences and their component evidentiary assertions and the presuppositional relations. Table 2, using the output of the Stanford Parser (a list of collapsed typed dependencies lacking the semantic information) for sentence [3] in the above passage from the case opinion, suggests how the presuppositional relation Causal-chain-assertion might annotate the dependencies relevant to the second Althen condition (see assertion MIN[4 of 4] at the bottom of Figure 2).

Table 2. Parser Output with Presuppositional Relation, Showing Relevant Dependencies (Shaded)

<table>
<thead>
<tr>
<th>Presuppositional Relations</th>
<th>Type dependencies, collapsed</th>
</tr>
</thead>
<tbody>
<tr>
<td>3 det(result-3, a-2)</td>
<td>prep.as(began-14, result-3)</td>
</tr>
<tr>
<td>3 prep.within(began-14, weeks-7)</td>
<td>poss(vaccination-11, her-9)</td>
</tr>
<tr>
<td>3 nn(vaccination-11, varicella-10)</td>
<td>prep.of(weeks-7, vaccination-11)</td>
</tr>
<tr>
<td>3 nsubj(began-14, petitioner-13)</td>
<td>xsubj(experience-16, petitioner-13)</td>
</tr>
<tr>
<td>3 root(ROOT-0, began-14)</td>
<td>aux(experience-16, to-15)</td>
</tr>
<tr>
<td>3 xcomp(began-14, experience-16)</td>
<td>det(onset-18, the-17)</td>
</tr>
<tr>
<td>3 dobj(experience-16, onset-18)</td>
<td>prep.of(onset-18, symptoms-20)</td>
</tr>
<tr>
<td>3 poss(encephalomyeloneuritis-23, her-22)</td>
<td></td>
</tr>
<tr>
<td>3 prep.of(symptoms-20, encephalomyeloneuritis-23)</td>
<td></td>
</tr>
</tbody>
</table>

(3) Relation extraction. DeepQA performs relation extraction on the PAS of sentences [7:2], and identifies domain-specific semantic-level relations in a sentence, together with the predicate-arguments (typically, typed entities) for those relations [4:10]. An example of a significant relation in playing Jeopardy! is “authorOf,” and a pattern of arguments might be “[Author][WriteVerb][Work]” [4:10]. Relation detection can be accomplished by hand-creating rules that capture such patterns, or by statistical approaches using training data [7:2-11]. For constructing legal arguments, new relations are needed in addition to those useful in question-answering. For example, the presuppositional relation Causal-chain-assertion (Table 1) takes as predicate-arguments Specific-vaccination and an instance of a Generic-injury. Another example is the DLF annotation for horizontal relevance, such as relevance to the second Althen condition (see Figure 2). In general, sentences containing causal relations whose arguments are proper nouns are more likely to be relevant to Althen conditions 2 or 3, and not to Althen condition 1.

(4) Knowledge base. The principal functions above can be applied to generate knowledge bases that are useful for performing specified tasks. For the purpose of playing the question-answer game of Jeopardy!, the knowledge base was a “bag of frames,” where a frame (consisting of pairs of slots and
values) represented relations and entities in a segment of text [1:2]. Frame projections are the portion of a frame that occurs with regularity in many frames and is of particular interest (e.g., for question-answering or argument-building). For example, the portion of a frame that includes only verb and object (i.e., V-O projection) is particularly useful for analyzing the types of objects associated with particular verbs in the dataset [1:5]. A domain-specific legal knowledge base would include the general knowledge of IBM’s PRISMATIC as well as the presuppositional and DLF information specific to a particular area of decision documents (e.g., VACCINES). The knowledge base contains the multiple frame projections that map into a presuppositional relation. This is important since all of the ways a given presuppositional relation can be realized in different sentences should map onto the same relation.

It may be helpful to picture the integration of the DeepQA analysis with the presuppositional and DLF analyses as a layering of the legal annotations (semantic-pragmatic) over the DeepQA annotations (syntactic-semantic). Figure 3 shows an argumentation-relevant sentence as input at the bottom.

Figure 3. Layering Legal over DeepQA Annotation

The first layer of legal annotation above the DeepQA annotation would be the semantic-pragmatic presuppositional annotation – using semantic relations such as those listed in Table 1. For example, this instance of a Subject-Verb-Object frame projection would be annotated as a Causal-chain-assertion in the V/IP Corpus.

The second layer of legal annotation comprises the DLF annotations. These annotations provide the propositional tree structure as a pragmatic layer, while the presuppositional and DeepQA annotations provide the syntactic-semantic-pragmatic sub-propositional structure. That is, the DLF annotations are useful for organizing sentences into lines of default-logic reasoning, while the presuppositional and DeepQA annotations provide a syntactic-semantic structure internal to those sentences. In Figure 3, this Causal-chain-assertion is annotated as being horizontally relevant to the second Athen causation condition.

7. SCORING DEGREES OF CONFIDENCE

One of DeepQA’s overarching principles is pervasive confidence estimation. DeepQA’s analytics score candidate answers and supporting pieces of evidence for the degree of confidence in their relevance to answering a question [2:4]. For this purpose Watson employs a large number of evidence-scoring strategies based on a range of syntactic and semantic information. In addition, based on a training corpus of question/answer pairs, DeepQA also learns to assign weights to the evidence-scoring strategies for different answer types.

We propose using a DeepQA approach to learning and confidence-scoring, but with the goal of finding argumentation-relevant sentences, extracting assertions (information) to use as evidence, and constructing legal arguments for and against the proposition that a new situation is an instance of compliance or non-compliance with a legal standard. Using an annotated corpus of decision texts, such a system could discover patterns associated with successful or unsuccessful arguments that include (for example) Causal-chain-assertions that a Specific-vaccination caused an instance of a Generic-injury (see Table 1 & Figure 3). The system could then use those patterns to assess the likelihood of relevance of a sentence to an argument, and the level of confidence in constructing a successful argument. For instance, confidence scores would reflect the likelihood that the information in a sentence includes relevant presuppositional relations, matches assertions in corpus cases, and fits horizontally or vertically into identified DLF argument structures. The annotation of DLF and presuppositional information in decision texts might also enable a DeepQA system to learn to assign weights to these evidence-scoring strategies, in order to score confidence in overall assessments of argumentation-relevance.

8. CONCLUSION

The V/IP Corpus provides critical resources for adapting DeepQA tools to legal analysis and for training and testing new algorithms for the combined NLP and legal analyses. Whereas IBM’s Watson project extracted training data from Wikipedia and DBpedia [7], we use the V/IP Corpus to provide training data. Functioning as a ground truth, the presuppositional and DLF annotations provide the means for developing relevance measures to score new documents, and for estimating the likelihood that they contain information relevant to argumentation in new legal cases. They also help in identifying and extracting argumentation-relevant sentences, extracting assertions (mining information) to use as evidence, and constructing legal arguments for and against the proposition that a situation is an instance of compliance or non-compliance with a legal standard.

9. REFERENCES