Automatic Recognition of Distinguishing Negative Indirect History Language in Judicial Opinions

Jack G. Conrad
Research & Development
Thomson Legal & Regulatory
St. Paul, Minnesota 55123 USA
Jack.Conrad@WestGroup.com

Daniel P. Dabney
West Online Research
West Group
St. Paul, Minnesota 55123 USA
Daniel.Dabney@WestGroup.com

ABSTRACT

We describe a model-based filtering application that generates candidate case-to-case distinguishing citations. We developed the system to aid editors in identifying indirect relationships among judicial opinions in a database of over 5 million documents. Using a training collection of approximately 30,000 previously edited cases, the filter application provides ranked sets of textual evidence for current case law documents in the editorial process. These sets contain judicial language with a strong probability of containing distinguishing relationships. Integrating this application into the editorial review environment has greatly improved the coverage and efficiency of the work flow to identify and generate new distinguishing relationship entries.

Keywords

legal research, case law analysis, citation loci, distinguished cases

1. INTRODUCTION

Parsing natural language text such as case law documents for complex, multifaceted concepts such as distinguishing relationships is a difficult task.[1] Yet modern legal researchers analyzing case law opinions regard such negative indirect history citations as indispensable. Today's online legal citator services have provided varying degrees of such case-to-case citation relationships. In some instances, newer online citation tools such as *KeyCite* have had to extensively harvest such postings in a retrospective manner.[2] In this paper, we introduce *Dparse*, a model-based automatic filter for distinguishing language, developed and trained using over 20 high-level rules along with a wide variety of lower-level subrules. Through sufficient training and failure analysis, our recognizer retrieves candidate distinguishing relationships at production precision and recall levels significantly surpassing those obtained during our training and testing phases as we discuss later in the paper. The system is being applied as a computer-assisted database update tool for attorney-editors employed to expand coverage of indirect history postings for an online citator application. After having successfully harnessed Dparse to run against retrospective cases, we plan to apply the recognizer to prospective cases as well, conserving a substantial amount of human editorial time in the task of identifying and extracting the same distinguishing language.

The remainder of this paper is organized as follows: Section 2 provides background on case law history topics. Section 3 describes related work in natural language pattern recognition, whether filtering or extraction. Section 4 outlines our experimental methodology and the development of our model. Section 5 describes the corpus of legal data underpinning all of our efforts. Section 6 describes the environments in which the filter was developed and applied. Some internal components of the application are detailed in Section 7. Sections 8 and 9 discuss our experiments and results. The logistic regression model used to tune performance is explained in Section 10. Our conclusions and description of future work are presented in Sections 11 and 12.

2. BACKGROUND ON CASE LAW "HISTORY"

The American legal system is based largely upon judicial precedent. The statements of judges in their written and published decisions represent the law and establish, at least in principle, new or extended law that is to be followed in similar instances. It is thus essential that lawyers and judges have access to the entire body of judicial opinions, as well as how a given opinion interprets relevant related opinions in the same jurisdiction, or from a higher court, in order to determine what law applies under a specified set of circumstances.

Using an extensive team of highly trained attorney-editors, West manually adds abstracts and point of law summaries, in addition to recording special types of citations made to other cases. Such citations to other historical case law documents can implicitly impact the relevance of the cited judicial opinions in both a positive (agreement) or negative (disagreement) manner. The role of a citator database is to preserve and track such historical relations among cases. Online citation databases and services are of great interest to the legal profession because they provide a means of testing whether a case is still based upon solid law. The utility

[1]The "distinguishing" negative indirect history designation in a judicial opinion, identified by a judge or judge's surrogate author of an opinion, denotes a relationship between cases whereby one case is shown distinct from another by means of one or more contrasting features (illustrated in Table 1 and elsewhere).

[2]We define *postings* as recorded entries in a table of features of interest.

CIKM'01, November 5-10, 2001, Atlanta, Georgia, USA.

of such a service is thus highly reliant upon the extent to which it, is (a) current and correct, and (b) complete.

The editorial history task thus involves examining court opinions for language that affects one or more previous cases, then noting, for each such case, the prior case or cases affected. The subsequent decisions then become part of the history of the previous, cited case.[3] For example, the present court, (usually called the "instant" court) might reverse the decision of the previous court on an appeal, and its decision thereby becomes part of the history of the earlier case.

There are two chief kinds of history. *Direct* history involves cases in the same appellate chain as the current case. Thus the instant case may be part of the direct history of an earlier case via an appeal from an earlier decision. Indirect *history* involves cases in other appellate chains, which usually appear because they are cited by the judge or counsel as logical precedents, with which the judge will often agree (positive history), but sometimes will disagree (negative history). Examples of some common negative indirect history patterns are illustrated in Figure 1.

No.	History Designation
1.	"Overruled by . . . "
2.	"Reversed in part by ... "
3.	"Declined to follow by ... "
4.	"Holding limited by ... "
5.	"Disagreement recognized by ... "
6.	"Followed with reservations by . . . "
7.	"Superceded by statute as stated in ... "

Figure 1: Examples of Negative Indirect History

A distinguishing designation is considered mild treatment by negative history standards and is one of the most common negative history markers found.[4] This designation comprises as much as two-thirds of the negative indirect history postings in a system like *Shepard's*. Given that *Westlaw* indexes over five million case law documents, the number of distinguishing case occurrences in this corpus would be substantial. Because the number of contextual clues for this treatment is extremely large, the task of manual review would be daunting, and would likely require hundreds of thousands of hours of editorial analysis.

Automated assistance to citator staff has the potential to reduce their workload appreciably, provided that the results are reliable. Such an application would thus require high recall (90%+) and acceptable precision (50%-60%+). We emphasize the detection of indirect history over direct history, since the latter is typically detected in the earliest stage of editorial treatment. Work on identifying direct history has formed the subject matter of other papers [13, 1]. The primary focus of our present work is on negative indirect history, since from a legal research standpoint it is more important to have overturned, weakened, or questioned opinions revealed rather than have reaffirmed or respected decisions featured. To be useful, a filtering system must offer a ranked list of prospective indirect history (90% of the time or better); that is, true indirect history must be in the candidate list over 90% of the time. We would also like the correct indirect history postings to be ranked high on the list, to facilitate manual inspection.

[3] Note that 'history' in this sense extends into the future, not the past.

[4] Distinguishing treatment receives a yellow flag on West's *KeyCite* citator system.

3. PREVIOUS WORK

An appreciable body of work has focused on parsing supra-sentence natural language text with the goal of fostering sub-domain analysis such as citation analysis, if not extraction. Jacobs argues for a mutually-reinforcing syntactical and statistical approach to parsing, harnessing traditional linguistic analysis to identify relations, then extending the application by leveraging statistical, domain-driven processing in order to improve semantic results. He applies this technique to both word sense disambiguation and data extraction [14]. Using a related approach, Richardson uses a broad coverage rule-based parser to calculate probabilities while parsing an untagged corpus of natural language text. He then uses those statistics when applying the same parser on new text, while achieving improved performance [17]. Schweighofer et al. similarly argue that conventional IR systems fall short in legal applications because of their preoccupation with syntactic representations [18]. They have found that a more effective method is to represent legal knowledge using linguistic tools, statistics and neural networks in order to track relationships. They claim that this combination is more efficient than clustering, for example, since it eliminates the need for fine-grained text processing (e.g., stopping, stemming, etc.) and threshold selections. This network approach is viewed as a front end to a knowledge acquisition application. Sekine uses a relatively small set of test corpora representing eight different domains to determine that parsing performance is best when training in the same domain and that performance degrades monotonically when one moves away from the domain to (in order) the same class of domains, combined domains, an alternative class and lastly, an alternative domain [19]. Kessler et al. leveraged similar lexical clues or *facets* to distinguish successfully between an assortment of genres. They concluded that surface clues can assist one in the determination of genre, which can in turn aid in analyzing deeper structural properties [15].

Research has also been directed towards the extraction of certain well-defined features from textual databases. Goldberg et al. rely on user collaboration to filter out certain types of e-mail [11]. By contrast, Bochereau et al. invoke a multilayered neural network to extract decision rules from a legal corpus of 50,000 cases, claiming that only 1,000 "logical clauses" are required to train for a given rule [4]. Others have paused to develop ontologies in an effort to better characterize the building blocks of the legal landscape [7, 3]. Yet relatively little has addressed a means to identify automatically opinion history, especially negative, not to mention frequently occurring distinguishing references in case law documents.

The study of case law history designations falls generally under the topic of citation analysis. Much has been written about citation analysis in the domain of scientific articles; relatively little has been reported on the science of legal citation analysis. Jackson et al. have developed a technique to identify and verify citations in the prior case (positive) history space [13, 1]. They developed ***History*** Assistant, an information extraction and retrieval system that extracts rulings from court opinions and retrieves relevant, prior cases from a citator database. The application uses MUC-like technology [2], but requires stronger pars-

ing methods in order to consider multiple interpretations of sentence fragments, and links new documents to related documents that may be affected. This work focuses on cases in the same appellate chain, however, and does not address the broader spectrum of negative indirect history.

Case-based reasoning approaches have contributed to the scope of knowledge on these problems. Golding and Rosenbloom discovered through research in the non-legal domain that case-based reasoning in tandem with a rule-based system produces results superior to rule-based systems alone [12]. Brüninghaus developed DANIEL, an expert system integrating case-based and rule-based reasoning. The two components work autonomously and concurrently and resolve conflicts through a separate coordination component [5]. By contrast, Daniels and Rissland integrate IR and CBR approaches in SPIRE, a system that first retrieves relevant documents from a large collection of texts and then highlights passages containing relevant information about legal issues of interest [9, 10].

Cohen and Kudenko examine an approach to alleviating poor filter performance during training by collaborating with filters learned from other users. They find the method robust against a variety of users and learning techniques involved in training classifiers for the purpose of filtering [6]. Spertus developed a dedicated filter for the purpose of flagging abusive ('flame') mail in an inbox [20]. It operates a rule-dependent five-stage architecture. It is characterized by a high incidence of both misses and false positives. Although brittle, it reflects the nuances and subjectivity of what is often non-literal language. The analogy is limited, but Spertus' "Smokey" system is similar in function, at a sufficiently high level of abstraction, to our own task, which consists of (1) identifying special relationships through the analysis of important language clues, both positive and negative, (2) iteratively training a system on those clues, and (3) tuning the system based on a thorough analysis of misses and errors.

4. METHODOLOGY

In order to focus on the distinguishing components of case law documents, we developed a model-based study. One of the lessons surfacing from the approaches discussed above is the utility of integrating more than one NLP technique (e.g., rule-based & case-based [12, 5], retrieval & case-based [9,10], retrieval & extraction-based [13,1], etc). Our method relies upon a similar strategy, one that harnesses a retrieval/extraction module after the completion of a rule-based system; it is additionally supported by a logistic regression rule-ranking function that is applied to the evidence satisfying our rules. This method was developed by means of a detailed examination and analysis of thousands of individual case patterns. Our development methodology is described below.

4.1 Initial Rule Generation

Preliminary rule candidates were compiled from recent cases in which editors were asked to identify and record distinguishing relationships and their characteristic language. This focus was responsible for roughly one-third of rules that we eventually needed to accommodate.

4.2 Alpha-Beta Rules

When a judicial opinion distinguishes itself from another judicial opinion, the distinguishing relationship is between the instant case and a case it is directly citing. We call this a primary alpha-beta-type relationship $[\alpha \Rightarrow \beta]$. Constructions like the following are examples of $(\alpha \Rightarrow \beta)$ relationships: "The instant case [A] is distinguishable from case [B] for . .." and "Case [B] is not controlling with respect to case [A] since conditions . . . " Over 90% of the relationships recognized by *Dparse* fall into this primary category.

4.3 Beta-Gamma Rules

By contrast, when a judicial opinion is discussing another legal case which in turn is distinguishing itself from yet another legal case, the opinion is referring to a distinguishing relationship between two separate cases. We call this indirect citation reference a secondary or beta-gamma-type relationship $[\alpha \Rightarrow (\beta \Rightarrow \gamma)]$. Patterns like the following are illustrations of $(\beta \Rightarrow \gamma)$ relationships: "In case [B], the court shows its facts distinct from those in case [C] because . .." and "Case [C] was found inapposite in relation to case [B] as a result of . . . " Although the postings contributed by $(\beta \Rightarrow \gamma)$ relationships amount to only a fraction of our overall pool, they occasionally reveal distinguishing relationships we would not have otherwise identified.

4.4 Editorial Advisory Group

Our work was additionally supported by a team of five attorney-editors working in an editorial department drafting abstracts and point of law summaries. These editors would normally edit approximately 40 cases per week and had a combined total of 42 years of editorial experience. Before West editors began routinely identifying distinguishing case postings during the editorial process, these five were assigned the task of monitoring the forms distinguishing relationships would take. We met monthly for roughly three months to discuss the language patterns and variations in distinguishing relationships. This process was an invaluable resource that facilitated the creation and growth of a knowledge base of distinguishing relationship types.

4.5 Empirical Rule Generation

Based on our processing of a large training collection in which correct negative history information was available, we were able to deduce numerous supplemental rules or subrules from the evidence. Editor-generated assignments from this collection helped reveal additional language nuances, much of which contributed to an overall increase in filter recall.

4.6 Failure Analysis Review

A second team of attorney-reviewers aided us in our review of system misses or failures. Once Dparse was run on our training data, we asked three attorney-reviewers to examine and characterize the filter's misses and incorrect assignments. This iterative feedback process was responsible for numerous system improvements, usually in the form of new rules, or tighter existing rules with additional acceptance or rejection criteria. This step thus provided performance improvements in terms of both recall and precision.

5. DATA

West Publishing, a West Group entity, first began publishing judicial opinions in the 1870s, and its National Reporter

System (NRS) now contains approximately five million published opinions from virtually every state and federal jurisdiction. A typical pre-edited case contains no significant markup. It includes a title with parties, court, date, and other information, and is about seven pages long, although some cases can be 50 or more pages long. The main body of a case is its opinion-of-the-court section, which often begins with a statement of how the case came before the court. It may then present a recapitulation of the facts, before discussing and analyzing one or more points of law. Finally, it will usually contain one or more rulings, such as "we grant the plaintiff's motion for a new trial" or "the decision of the trial court is affirmed." Depending on one's level of granularity, there are between 500 and 2,000 different rulings a court can make. It is significant to note that different jurisdictions tend to organize their opinions differently, and an individual judge or panel of judges has great latitude in determining the structure and content of any opinion their court authors. Differences in language and writing styles from one opinion to another can be profound.

West attorney-editors began to record the distinguishing case relationships that they encountered in August of 1997. We relied upon the next six months' worth of incoming legal cases, fully examined for distinguishing relationships, for our training sets. Our baseline training set of cases was received and edited between mid-August and mid-November, 1997. These consisted of 31,247 case law documents and contained 2,396 recorded distinguishing case relationships. Our baseline *test* set of cases was received and edited between mid-November, 1997 and mid-January, 1998. These consisted of 7,181 case law documents and contained 1,993 recorded distinguishing case relationships.

Collection	No. Docs	Distinguishing Postings
Training	31,247	2,396
Test	7,181	1,993

Table 1: Collection Statistics

6. SYSTEM ENVIRONMENT

Dparse was developed using a *flex*-based lexical pattern recognizer designed and developed in our lab, initially to detect sentence boundaries and quoted materials, but subsequently to tokenize more general natural language constructs. It was developed first on our Sun Enterprise 4500 server with 3 CPUs and was then ported to production-side mainframes. Parallel processes were run on cases in batch mode, one data component (i.e., physical database) per process. Approximately 5 GB of data were treated per 12-hour period. The process bottleneck, if there was one, occurred during the loading of filter-certified evidence to our relational citator database. This task was usually completed the following day.

7. RULE-BASED MODEL DEVELOPMENT

Our initial rules were provided by legal domain experts assigned to coordinate our project advisory group and failure analysis team. Additional rules were recommended through our interaction with these groups. We considered these suggestions and cautiously invoked rules which were neither too broad (low precision, going beyond distinguishing) nor too esoteric (low recall).

In developing these rules in a natural language framework, we discovered that we were able to tune our performance in a number of important ways. First, non-case law citations and pro forma history citations are removed from consideration when applying our rules.[5] Secondly, evidence occurring within quotations is usually not considered by the algorithm. Such material is ignored because it is coming from other cases or law and not the instant case. In addition, text containing parenthetical or bracketed expressions which separates distinguishing evidence from citations in proximity to them are not considered in the applicable windows. The inspection windows described below were determined empirically. Also, distinguishing evidence which has negative language markers near it is more closely inspected to determine if the given construction is not a negative one. Furthermore, when citations under discussion are made to statutes or code, the candidate evidence is often removed from consideration based on some of the heuristics outlined below.

7.1 Analysis Functions

The following functions illustrate a number of the analysis tools Dparse may apply when examining judicial language.

- neg_nearby( ) – probes for language of negation in proximity, e.g., "insufficiently", "inadequately", "fails."
- disagreenearby( ) – probes for language of disagreement, e.g., "no/without merit", "we/must/strongly disagree", "do not/cannot agree."
- code-rulenearby( ) probes for "stat.", "code", "rule", "U.S.C." or statute SECTION tag.
- statute_nearby( ) – looks for "statute(s)."
- distinguish_check$_b$( ) – inspects (backwards) for disputative verbs, keywords, e.g., "incapable of", "contends", "insists" , "refuses."
- distinguish_check$_f$( ) – inspects (forwards) for adjectives for use with judges, counsel, etc., e.g., "justice", "colleague."
- subject_nearby( ) – determines whether 'distinguished' is used as verb modifying preceding subject, e.g. "the officer was distinguished in his service."
- diminish_nearby( ) probes for weakening language, e.g., "mere", "weakened", "reduced," etc.

7.2 Citation Surrogates

The application is capable of recognizing surrogates for citations, for instance, anaphoric references to previously made citations, so that the lack of a bona fide citation will not by itself prevent a rule from being satisfied.

The take-home value of these heuristics is that we want to pay close attention to that which is important evidence; conversely, we do not want to focus too much attention on that

[5] *Pro* **Forma history citations are presented in a case for the sake of form and convention, made to broadly applicable boilerplate law that usually contributes little to the details of the case in question.**

Rule No.	Sub-rule	Rule-type	Lexical and Citation-related Clues
0	(a)	[$\beta \Leftrightarrow \gamma$]	[cite β] contra/but cf./but see [cite γ]
	(b)	[$\alpha \Leftrightarrow \beta$]	Like (a), "but" w/ single [cite]
1	(a)	"	distinguish*, distinction(s) (present tense)
	(b)	"	distinguished (past tense)
2	(a)	"	inapposite, inapplic*, irrelevant, impertinent
	(b)	"	not appli*, no appl*, no govern*, not control*, not decisive/relevant/pertinent/binding
	(c)	[$\beta \Leftrightarrow \gamma$]	2 (b) w/ was not, did not, or had not
3	(a)	[$\alpha \Leftrightarrow \beta$]	dissimilar*, differ*, divergen* /n fact(s)/circumstance(s)/situation(s)/issue(s)/purpose(s)
	(b)	[$\beta \Leftrightarrow \gamma$]	involv* /2 different, difference /1 between
4	(a)	[$\alpha \Leftrightarrow \beta$]	unlike /n [cite], unpersuaded /n [cite]
	(b)	"	unlike /n plaintiff(s)/defendant(s)/claimant(s), unpersuaded /n plaintiff(s)/defendant(s)/ ...
5		"	not /n/ same /m fact(s)/issue(s)/circumstance(e)/situation(s) ...
6		"	reliance upon [cite] not supported/not well placed/ ... is misplaced/misguided/misleading/ ...
7	(a)	"	in/by contrast /s fact(s)/issue(s)/circumstance(s)/situation(s)/purpose(s) ...
	(b)	"	in/by contrast /s [cite]/[cite surrogate]
8		"	compare/cf. +3 [cite]
9		"	does not/cannot help/support/inform/aid/persuade ...
10		"	not persuaded/convinced +s [cite]
11		"	not +2 all +2 fours with ...
12	(a)	"	conversely/whereas/rather /6 [cite]
	(b)	"	However /6 [cite]

Table 2: Breakdown of Dparse Rules (examples)

which is ultimately less significant. We present a summary of our high-level rules in Table 2.[6]

8. EXPERIMENTS

We performed two iterations of training-test collection runs using our expanded rule set. In each instance, we trained on a set of manually reviewed cases for which we had nearly complete knowledge of the distinguishing postings present. We then ran our algorithm on a new test set of data that our algorithm had previously not seen. The experiments we report on here involve our second round of training and testing.

We used 2,965 Editor-identified postings for training set II (composed of 1,660 postings from Training I and 1,305 postings from Test I) from 1,513 cases (831 from Training I and 682 from Test I) receiving the distinguishing treatment code as the standard. These figures reduce to 2,730 when postings from unreported cases are removed from this set (since the filter will generally not be used against this set). This further reduces to 2,453 when 1,120 double counts from two special federal collections are removed, and finally to 2,396 when the few spurious non-Training II cases are purged.

9. RESULTS

The results of our experiments conducted on our large phase II training and test data are presented in the Appendix. Table 4 compares the differences in precision and recall from our training and test sets. It should be pointed out that each of the rules' precision values and just under half of the rules' recall values from our smaller test set surpass those obtained from our larger training set. This development suggests that the coverage produced by our training set is fairly comprehensive. Also worth noting is that the training set was reviewed and generated by our editors several months before they examined our test set. We thus infer that the 40 editors working on the task were themselves better trained and acquainted with the variations of the distinguishing evidence by the time they treated those cases. There is a fairly wide range of precision values associated with these rules, however, which may serve to emphasize the scope of the false positives that our rules, tuned as they are, could still produce.

Actual precision and recall figures in our production environment are substantially higher than those from either our training or testing phases. The result of our dedicated training and failure analysis is that *Dparse* recognizes candidate distinguishing relationships at production recall levels exceeding 80% with production precision levels approaching 50%. One of the reasons for this discrepancy is that in our experimental environment, we evaluated performance based on the assumption that our editorial coverage was (a) complete and (b) correct. We thus penalized our experimental version for absent postings (misses) as well as erroneous postings (false positives), even when *Dparse* correctly identified postings missed by the editors or which fell into one of the other negative history categories (Figure 1). These two conditions were treated strictly as errors. We have discovered in our production environment, however, that editorial misses do occur, however infrequently. The additional postings that Dparse generates in these situations help rectify what was effectively a lower bound performance measured through our testing phase. As a result, our actual production precision and recall figures are much closer to our target performance values outlined in Section 2 than to those obtained in our development phases.

[6] /n and +n represent unordered and ordered proximity operators, respectively, whereas /s means in the same sentence.

10. LOGISTIC REGRESSION RANKING MODEL

Based on results from our test data, we are able to determine which rules have the highest probability of producing correct distinguishing postings (i.e., the highest precision rules). We use a logistical regression ranking model to determine a performance-based ordering of the most effective rules [16]. A model derived by logistic regression can be used to combine widely varying and otherwise problematic contextual clues into a single estimate of the relevance of candidate citations [8]. We use our estimates derived in this way as our system's ranking function.

Our application uses logistic regression to meld a variety of highly-dependent relevance clues into a single probability estimate. The clues considered include several different coefficients of associated evidence taken from our standard Dparse output matrix.

We are interested in a model that uses a single relevance indicator, one that can take multiple parameters and that can be associated with measurable likelihoods. In such a model, the likelihood ratios show how to make the relative contributions of each piece of evidence apparent. Using Bayes Rule and an assumption of linked evidence dependencies, the posterior odds of relevance (i.e., distinguishing citations) for n indicators is calculated and maximized:

$$\frac{P(Rel|x_1,...,x_n)}{P(\neg Rel|x_1,...,x_n)} = \frac{P(Rel)}{P(\neg Rel)} \cdot \prod_{i=1}^{n} \frac{P(x_i|Rel)}{P(x_i|\neg Rel)}$$

In addition, the process of building the model sheds light on the usefulness of individual clues, and identifies a subset of rule-specific clues that contain practically all of the non-redundant relevance information of the complete set. For this reason, not all of our composite set of rules appear in Table 3, which contains the top-ranked rules determined via logistic regression. It represents the order in which the attorney-reviewers are presented evidence for the candidate postings.

#	Rule No.	Coeff	#	Rule No.	Coeff
1.	Rule 1 (a)	1.096	8.	Rule 10	0.352
2.	Rule 6	0.876	9.	Rule 7 (b)	0.261
3.	Rule 4 (a)	0.605	10.	Rule 4 (b)	0.132
4.	Rule 5	0.482	11.	Rule 3 (a)	0.132
5.	Rule 7 (a)	0.439	12.	Rule 2 (b)	0.132
6.	Rule 2 (a)	0.433	13.	Rule 12 (a)	0.105
7.	Rule 1 (b)	0.396	14.	Rule 12 (b)	0.100

Table 3: Ranked Rules via Logistic Regression

If one examines the rules which do not appear in the table-8, 9, 11 [$\alpha \Leftrightarrow \beta$ rules] and O(a), 2(c), 3(b) [$\beta \Leftrightarrow \gamma$ rules]-one sees rules that are (i) associated with low recall, (ii) pointers to separate $\beta \Leftrightarrow \gamma$ pairs, or (iii) have been largely covered by parallel evidence surfacing through the other rules. Because of this last redundancy factor, the logical regression process was useful in promoting the richest and most efficient performers to the top of the list.

11. CONCLUSIONS

Dparse is a model-based approach to automatic recognition of negative indirect history in court opinions; it relies upon lexical clues supplied both by domain experts and our training collection. It has demonstrated that it is capable of analyzing case law documents and supplying this history generating language with high recall and satisfactory precision. Given that we discounted the non-distinguishing but alternative citation candidates, labeling them as failures, the actual performance of our filter for negative indirect history actually surpasses the results reported here by at least an estimated 10%. The filter's output is nonetheless used in a post-process manual review framework. The economy of editorial review is obtained from the fact that editors do not read the opinion text of every case; rather, they focus their attention on the highly distilled evidence from the application. The results in Section 9 indicate that, on average, they will discover one or more additional negative indirect history posting per candidate window.

Comparing our work with other efforts in the literature, we think it unlikely that a machine learning approach would have achieved acceptable precision when faced with the myriad permutations of opinion text which is far less concise and transparent than, for instance, news text, and much more complex than average e-mail correspondence. We also think it improbable that court opinions would lend themselves successfully to statistical modeling alone, given the high degree of variance between court documents from different jurisdictions, not to mention authors. Although statistical modeling in IR and IE is advancing, we opted to harness a more proven MUC-like approach to attain our goal.

Dparse is clearly a domain-specific application, and not a tool to be easily applied to general-purpose needs. Its parser, means of generation, and training all focus on the task of history extraction from case law documents. We have, however, found that the internal components of Dparse are robust enough to offer useful building blocks for other essential NLP tasks, including sentence boundary detection, quotation identification, and the recognition of assorted specialized phrases.

As for the application itself, at the time of this writing, it has been responsible for the identification of nearly 500,000 distinguishing postings. Our enterprise has employed roughly a dozen legal professionals to examine the output from the citator database system which Dparse feeds. In little more than twelve months, the legal team verified over 450,000 distinguishing cases relationships, not to mention tens of thousands of related postings (Figure 1). And this quantity was generated by only about one-half of the total relevance-ranked evidence produced by the filter and available in our citator database.

12. FUTURE WORK

In order to demonstrate that our distilled set of rules is generalizable, we are in the process of applying ***Dparse*** to other distinct but affiliated domains. Beyond case law, these domains include secondary law (Law Reviews), NLRB (Labor Relations), Norton (Bankruptcy), Couch (Insurance), and Rutter (California Practice Guides), to name just a few. It is not unusual for such expanded use to uncover new language patterns and thus to warrant new rules. In a few of the above instances, this is precisely what was required. We are currently in the process of determining if our performance and rule rankings hold up in these other domains.

13. ACKNOWLEDGEMENTS

We thank Dan Gannon for contributing numerous lexical patterns for rule candidates and his many hours of evaluating the preliminary output of *Dparse*. We are also grateful for his coordination of our failure analysis team. We thank Chip Allen, Roberta Borchardt, Andy Martens, Nick Koster and George Westlund for proposing numerous additional candidate rules resulting from their extensive review of opinions for distinguishing language. We appreciate the assistance that Ted Cabbage, Laura Clements, and Ann Kennedy provided through their elaborate analysis of recognizer failures. Credit also goes to Bokyung Yang for her help with data acquisition, citation analysis, and preliminary Dparse interface design. Finally, we are very grateful to Bob Haschart for developing the *flex*-based tokenizer used in *Dparse* and for his constructive functional support throughout the project.

14. REFERENCES

[1] Khalid Al-Kofahi, Brian Grom, and Peter Jackson. Anaphora resolution in the extraction of treatment history language from court opinions by partial parsing. In *Proceedings of the* Seventh International Conference on Artificial *Intelligence* and Law *(ICAIL-99), Oslo,* Norway, pages 138-146. ACM Press, June 1999.

[2] ARPA. Proceedings *of the Sixth ARPA* Message Understanding Conference *(MUC-6)*, San Mateo, CA, 1996. Morgan Kaufman.

[3] Trevor J.M. Bench-Capon and Pepijn R.S. Visser. Ontologoes in legal information systems: The need for explicit specifications of domain conceptualizations. In *Proceedings of the* Sixth International Conference on Artificial Intelligence and *Law (ICAIL-97),* Melbourne, Australia, pages 133-141. ACM Press, June-July 1997.

[4] Laurent Bochereau, Danièle Bourcier, and Paul Bourgine. Extracting legal knowledge by means of a multilayer neural network application to municipal jurisprudence. In *Proceedings of the* Third International Conference on Artificial *Intelligence* and Law *(ICAIL-91),* Oxford, England, pages 288-296. ACM Press, June 1991.

[5] Stefanie Briininghaus. Daniel: Integrating case-based and rule-based reasoning in law. In Proceedings *of the Twelfth* National Conference on Artificial Intelligence *(IAAI-94),* page 1428. AAAI Press/The MIT Press, August 1994.

[6] William Cohen. Transferring and retraining learned information filters. In Proceedings *of the Fourteenth* National Conference on Artificial Intelligence *(IAAI-97)*, pages 583-591. AAAI, AAAI Press/The MIT Press, June-July 1997.

[7] Jack G. Conrad and Daniel P. Dabney. A cognitive approach to judical opinion structure: Applying domain expertise to component analysis. In Proceedings *of the Eighth International* Conference *of* Artificial *Intelligence* and Law *(ICAIL-01) St.* Louis, *MO,* pages l-11. ACM Press, May 2001.

[8] Daniel P. Dabney. Statistical Modeling *of* Relevance Judgments *for* Probabilistic *Retrieval of* American Case Law. PhD thesis, University of California at Berkeley, Library and Information Studies, 1993.

[9] Jody J. Daniels and Edwina L. Rissland. Finding legally relevant passages in case opinions. In *Proceedings of the Sixth* International Conference on Artificial Intelligence and *Law (ICAIL-97),* pages 3946. ACM Press, June-July 1997.

[10] Jody J. Daniels and Edwina L. Rissland. Integrating ir and cbr to locate relevant texts and passages. In *Workshop* on Legal Systems, Toulouse, France, Sept 1997.

[11] D. Goldberg, B Oki D. Nicholas, and D. Terry. Using collaborative filtering to weave an information tapestry. Communications *of the ACM,* 35(12):61–70, December 1992.

[12] Andrew R. Golding and Paul S. Rosenbloom. Improving rule-based systems through case-based design. In Proceedings *of the* Ninth International Conference on Artificial Intelligence *(AAAI-91),* Anaheim, *CA,* pages 22-27. AAAI Press, July 1991.

[13] Peter Jackson, Khalid Al-Kofahi, Chris Kreilick, and Brian Grom. Information extraction from case law and retrieval of prior cases by partial parsing and query generation. In Proceedings *of the* Seventh International Conference on Information and *Knowledge* Management *(CIKM-98), Bethesda, MD,* pages 60-67. ACM Press, November 1998.

[14] Paul S. Jacobs. Parsing run amok: Relation-driven control for text analysis. In Proceedings *of the Tenth* International Conference on Artificial Intelligence *(AAAI-92)*, pages 315-321. AAAI Press/The MIT Press, July 1992.

[15] Brett Kessler, Geoffrey Nunberg, and Hinrich Schutze. Automatic detection of text genre. In Proceedings *of the 35th* Annual Association for *Computational* Linguistics *(ACL-97),* Madrid, Spain, pages 32-38. Morgan Kaufman, July 1997.

[16] Marija J. Norušis. *Logistic* Regression Analysis, pages l-30. SPSS, Chicago, IL, 1994.

[17] Steve Richardson. Bootstrapping statistical processing into a rule-based natural language parser. In *The* Balancing *Act:* Combining *Symbolic* and Statistical *Approaches to* Language: Proceedings *of the Workshops (ACL),* pages 96-103, Las Cruces, NM, July 1994.

[18] Erich Schweighofer, Werner Winiwarter, and Dieter Merkl. Information filtering: The computation similarities in large corpora of legal texts. In Proceedings *of the* Fifth International Conference on Artificial *Intelligence* and Law *(ICAIL-95), College* Park, *MD,* pages 119-126. ACM Press, May 1995.

[19] Satoshi Sekine. The domain dependence of parsing. In Proceedings *of the* Ninth *Conference* on Innovative Application *of Artificial Intelligence (IAAI-97),* Providence, *RI,* pages 96–102. AAAI Press/The MIT Press, July 1997.

[20] Ellen Spertus. Smokey: Automatic recognition of hostile messages. In *Proceedings of the* Ninth Conference on Innovative *Application of* Artificial *Intelligence (IAAI-97), Providence, RI,* pages 1058–1065. AAAI Press/The MIT Press, July 1997.

15. APPENDIX:

PERFORMANCE EVALUATION — PRECISION & RECALL —

What follows are the tabular results of the precision and recall performance assessments of the filter on both our 31,247 document training collection and document 7,181 test collection.

From **Training Collection II,** these figures represent results from ***Dparse* Rules O-12** run on the new baseline training collection consisting of 31,247 **cases. Recall** uses the 2,396 Editor-identified postings (from the 31,247 cases) that received the distinguishing treatment code designation.

From **Test Collection II,** these figures represent results from Dparse **Rules O-12** run on the new baseline test collection consisting of 7,181 **cases. Recall** uses the 1,993 Editor-identified postings (from the 7,181 cases) that received the distinguishing treatment code designation.

Rule Number	Revised Precision (Training)	Revised Precision (Testing)	Percent Change	Recall (Training)	Recall (Testing)	Percent Change
Rule 0 (b)	4.38%	6.99%	+59.6%	0.75%	0.81%	+8.0%
Rule 1 (a)	29.86%	34.86%	+16.7%	25.75%	21.86%	-15.1%
Rule 1 (b)	12.72%	15.07%	+18.5%	4.22%	4.03%	-4.5%
Rule 2 (a)	10.74%	13.09%	+21.9%	10.48%	9.22%	-12.0%
Rule 2 (b)	6.17%	9.45%	+53.2%	6.89%	7.30%	+6.0%
Rule 2 (c) [$\beta \Leftrightarrow \gamma$]	5.35%	6.30%	+17.8%	0.92%	0.76%	-17.4%
Rule 3 (a)	7.80%	9.78%	+25.4%	4.51%	3.88%	-14.0%
Rule 3 (b) [$\beta \Leftrightarrow \gamma$]	5.37%	16.50%	+207.0%	0.75%	1.56%	+108.0%
Rule 4 (a)	17.18%	24.80%	+44.4%	11.48%	11.13%	-3.0%
Rule 4 (b)	22.73%	24.62%	+8.3%	1.04%	0.81%	-22.1%
Rule 5	5.88%	18.60%	+216.0%	0.08%	0.04%	-50.0%
Rule 6	23.16%	36.92%	+59.4%	1.84%	2.42%	+31.5%
Rule 7 (a)	16.49%	18.80%	+14.0%	1.29%	1.10%	-14.7%
Rule 7 (b)	11.99%	18.84%	+57.1%	1.71%	1.96%	+14.6%
Rule 8	1.88%	3.84%	+104.3%	0.67%	1.01%	+50.7%
Rule 9	6.99%	8.98%	+28.5%	4.63%	3.98%	-14.3%
Rule 10	10.00%	13.33%	+33.3%	0.38%	0.40%	+5.3%
Rule 11	0.00%	100.00%	+∞	0.00%	0.10%	+∞
Rule 12 (a)	6.58%	8.37%	+27.2%	13.02%	10.73%	-17.6%
Rule 12 (b)	3.16%	5.15%	+63.0%	9.72%	11.28%	+16.0%
Total [*tl–ml pairs*]	8.72%	11.79%	+35.2%	100.00%	94.48%	-5.5%
Total [w/o Rule 12]	**13.86%**	15.93%	+14.9%	89.15%	72.55%	-18.6%
Total [uniq pairs]	6.32%	9.15%	+44.8%	62.23%	59.09%	-5.0%
Total [w/o Rule 12]	**9.60%**	**12.74%**	+32.7%	54.51%	50.83%	-6.8%
Avg Prec. & Recall	**15.94%**	**20.39%**	+27.9%	12.43%	10.83%	-12.9%

Table 4: Precision & Recall (Precent Change) — Training vs. Test Collection

The tl-ml pairs referred to in the table correspond to the citing-cited case 2-tuples present in our citator history database. Bona fide negative history postings other than distinguishing are removed from our scoring, thus producing the smaller 'revised' set of actual postings.