Legal Knowledge and Information Systems
S. Villata et al. (Eds.)

doi:10.3233/FAIA200857

Extracting Outcomes from Appellate Decisions in US State Courts

Alina PETROVA [1], John ARMOUR and Thomas LUKASIEWICZ
University of Oxford, UK

Abstract. Predicting the outcome of a legal process has recently gained considerable research attention. Numerous attempts have been made to predict the exact outcome, judgment, charge, and fines of a case given the textual description of its facts and metadata. However, most of the effort has been focused on Chinese and European law, for which there exist annotated datasets. In this paper, we introduce CASELAW4 — a new dataset of 350k common law judicial decisions from the U.S. Caselaw Access Project, of which 250k have been automatically annotated with binary outcome labels of AFFIRM or REVERSE by our hybrid learning system. To our knowledge, it is the first attempt to perform outcome extraction (a) on such a large volume of English-language judicial opinions, (b) on the Caselaw Access Project data, and (c) on US State Courts of Appeal cases, and it paves the way to large-scale outcome prediction and advanced legal analytics using U.S. Case Law. We set up baseline results for the outcome extraction task on the new dataset, achieving an F-measure of 82.32%.

Keywords. legal analytics, outcome extraction, legal reasoning, outcome prediction

1. Introduction

Legal analytics – the application of computational methods to legal materials – has recently become a topic of global research interest. It offers potential to improve access to justice, automate repetitive administrative tasks, reduce legal costs, and bring transparency to judicial procedures [4]. Considerable research effort has recently been devoted to case outcome prediction — the task of predicting the outcome of a court's decision in a particular case (i.e., verdict, sentence, charge, or fine) given the factual background of the case [13,12,17,22,24]. Legal analytics requires a sufficiently large-scale dataset of case information, including facts and outcomes. However, legal data is usually stored in textual form with limited metadata. In particular, the outcome of a case is rarely stated explicitly in the case report and has to be extracted from text manually or (semi-)automatically.

In this paper, we investigate the problem of large-scale outcome extraction from common law judicial decisions. We introduce CASELAW4 — a novel dataset of 350k U.S. decisions from state Courts of Appeal, sourced from the Caselaw Access Project [8] that are annotated with outcomes. The annotation has been done in part manually but primarily with a hybrid outcome extraction model that reaches an F-measure of 82.32%.

[1] Corresponding Author: Alina Petrova; E-mail: alina.petrova@cs.ox.ac.uk.

Both the annotated data and the model are publicly available, and they act as baselines for outcome extraction both for opinions from US state Courts of Appeal cases and for the U.S. case law more generally.

2. Related Work

2.1. Legal Information Extraction

The works on legal information extraction are limited, and they adopt techniques from general-domain NLP. CAIL2018 [10], the largest publicly available dataset on Chinese Criminal Law, spurred works in legal event extraction and named entity recognition [26,30]. Few works focused on extracting particular types of clause sentences, e.g., sentences containing statutory terms [27], confidentiality clauses,[2] or even outcome sentences.[3] Unfortunately the latter proved to work poorly on appeal outcomes. For U.S. court data, prior work on outcome extraction has been done manually or semi-manually using dockets of US Federal Courts [28,29].

2.2. Legal Outcome Prediction

Legal outcome prediction is one of the most actively researched tasks in legal natural language processing. Previous works focused mostly on European and Chinese law. They include predicting outcomes in the French Supreme Court [18], in the European Court of Justice [14,19], and in the European Court of Human Rights [13,12,15,16], as well as predicting outcomes of criminal cases from the Supreme People's Court of China [10,20,21,22,23,24,25]. However, very limited work focused on the U.S. and U.K. law systems [9,17], and to our knowledge, no attempt has yet been made to predict outcomes for cases from the CAP dataset [8].

2.3. RNNs and LSTMs

In this section, we motivate our choice of the machine learning algorithm that we used in Section 4 in order to train a baseline outcome prediction model on the CASELAW4 dataset. Textual documents, such as court proceedings and case reports, are a type of sequential data. Sequential inputs have two important properties: (1) they do not have fixed size, and (2) later input typically depends on earlier one. For example, a word at position t in a sentence may depend on various other words at positions $t-n$ and even at positions $t+m$, with $n,m>0$.

While in general deep learning models are successfully applied to natural language-related tasks, one type of deep learning models — recurrent neural networks (RNNs) — is specifically tailored to handle sequential input. Among various RNN architectures, long short-term memory (LSTM) models [1] perform particularly well, as they mitigate the problem of vanishing and exploding gradients in the network. The key component of an LSTM is the memory cell that contains self-recurrent connection as well as three gates (input, output, and forget), that regulate which information is kept in the cell, which

[2] https://github.com/LexPredict/lexpredict-lexnlp
[3] https://github.com/ICLRandD/Blackstone

is passed further, and which is ignored, respectively, while the model reads the input text word by word. Finally, bidirectional LSTMs (bi-LSTMs) [2] are a variation of LSTMs that read the input twice, in the original and in the reversed order, which allows them to take into account not only the preceding information, but the information further in time; this ability is typically beneficial when processing textual data. LSTMs and bi-LSTMs are considered to be the state-of-the-art models for numerous natural language processing tasks [5,6,7], including those in the legal domain [19,21,20]. It is reasonable to expect (bi-)LSTMs to efficiently capture key phrases and words that manifest legal outcomes in the appeal setting, such as *we reverse* or *is therefore affirmed*, hence we chose bi-LSTMs as baseline models for outcome extraction task (see Section 4).

3. Dataset

The Caselaw Access Project (CAP) is the largest publicly available dataset of U.S. court decisions [8]. It is maintained by the Harvard Law School. CAP consists of nearly 7 million case reports from all US state, federal, and territorial courts and covers the time period of 1658−2018. Each report contains metadata on the hearing, court of hearing, jurisdiction, judges and attorneys, as well as the full text of the court's decision. Each report typically contains a review of key facts and previous court rulings, the legal reasoning applied by the court, and the verdict; it may also contain corrections and dissenting opinions. The reports are in unstructured form, but occasionally may contain section headings, e.g., *Facts* or *Conclusion*.

We have used a subset of CAP, CASELAW4, that consists of over 350,000 court case reports from New Mexico, North Carolina, Illinois, and Arkansas Courts of Appeal. These Courts hear appeals exclusively from lower courts within their respective states, on matters of domestic state law. The data for these jurisdictions are freely downloadable from the CAP website.[4] An example of a case report from CASELAW4 is presented in Figure 1. Since each case in CASELAW4 appeals some lower court ruling, the possible outcomes of each case are as follows:

- the previous ruling is kept as is (AFFIRM);
- the previous ruling is changed/annulled (REVERSE);
- some parts of the previous ruling are kept and some are changed (MIXED);
- the appeal is dismissed (a type of AFFIRM).

We intentionally treat cases with a clear-cut decision (AFFIRM and REVERSE) separately from more complex ones (MIXED), as it is common to establish outcome prediction baseline results first on simpler cases and only then move on to more complex, non-binary ones [13,17], and we foresee this as an avenue for future work.

Table 1 summarizes the dataset statistics, and Table 2 shows the distribution of cases depending on their length (as measured by the word count of the main body of the case, without dissenting opinions). As can be seen from Table 2, the cases vary a lot in length. We assume that the length of a case report is a fair estimation of the case's complexity: shorter case reports tend to either reinstate the decision of the first instance court (AFFIRM) or to give a clear reason why the existing ruling should be reversed

[4] https://case.law/download/bulk_exports/20200604/by_jurisdiction/case_text_open/

Table 1. Overview of CASELAW4

	New Mexico	Arkansas	North Carolina	Illinois	**Total**
Number of cases	18326	59696	97583	182771	**358376**
Avg length	2471.67	1545.98	1114.71	1812.33	-
Median length	1940	1262	672	1413	-

Table 2. Number of cases per word count in CASELAW4

Case length	New Mexico	Arkansas	North Carolina	Illinois	**Total**
< 200	952	2225	28439	29466	61082
$200-500$	738	5725	13037	12290	31790
$500-1000$	2466	14628	19103	25928	62125
$1000-2000$	5288	21862	20362	52144	99656
$2000-5000$	7036	14230	14312	53333	88911
> 5000	1846	1026	2330	9610	14812

(REVERSE). On the other hand, longer case reports usually indicate that the decision of the judges is non-binary (MIXED) and includes multiple sub-orders, or that more complex legal reasoning is involved.

The data in CASELAW4 are stored in JSON format. In addition to the original metadata about the case name, date, court, judges, cases cited etc., we annotated *a subset* of the cases with the AFFIRM or REVERSE outcome label (see Section 4). Finally, 500 cases from the New Mexico Court of Appeals are manually annotated with the AFFIRM, REVERSE, or MIXED outcome label as well as with the outcome sentences (also in Section 4). The dataset is publicly available on GitHub.[5]

Figure 1. Example of a case from CASELAW4


```
"casebody": {
    "data": {
        "corrections": "",
        "attorneys": [
            "Counsel",
            "Paul J. Derania and Scott S. Furs
tiffs and Appellants.",
            "Incorvaia & Associates, Joel L. I
Ehrich Lenz for Defendant and Respondent Affirmed
Group.",
            "No appearance for Defendant and R
Jose Family Housing Partners, LLC."
        ],
        "judges": [],
        "head_matter": "[No. H038874.\nSixth D
2014.]\nJAMES C. HILL et al., Plaintiffs and Appel
FIRMED HOUSING GROUP et al., Defendants and Respon
sel\nPaul J. Derania and Scott S. Furstman for Pla
pellants.\nIncorvaia & Associates, Joel L. Incorva
Ehrich Lenz for Defendant and Respondent Affirmed
Group.\nNo appearance for Defendant and Respondent
ly Housing Partners, LLC.",
```

```
        "opinions": [
            {
                "text": "Opinion\nPREMO, J.\nPlaintiffs James
C. Hill and Dawn L. Hill as trustees under a revocable trust dat-
ed February 17, 1977 (the Hills), appeal from a postjudgment or-
der awarding contractual attorney fees to Affirmed Housing Group
(Affirmed) under Civil Code section 1717.\nWe find no abuse of
discretion and affirm.\nI. Factual and Procedural Background\nIn
the underlying suit, the Hills sued San Jose Family Housing Part-
ners, LLC (LLC), and Affirmed, a managing member of LLC, for al-
leged violations of a written easement agreement. LLC and Af-
firmed (collectively defendants) were jointly represented by the
law firm of Incorvaia & Associates at a bench trial on the Hills'
claims. In their joint trial brief, defendants argued that, under
Corporations Code section former 17101, Affirmed could not be li-
able for LLC's actions solely because of its status as a member
of LLC. Defendants also jointly argued that the easement could
not lawfully be enforced and, in any event, they had not violated
the easement.\nThe trial court ruled that Affirmed was immune
from suit under Corporations Code former section 17101, noting
that the Hills had presented no evidence to show that Affirmed
engaged in any conduct outside of its capacity as a member of
LLC. As to LLC, the court rejected the illegality defense and
concluded LCC had violated the easement agreement. Following en-
```


4. Outcome Extraction

The first step towards outcome prediction is to extract outcome labels from case reports. Unfortunately, the original CAP dataset does not formally store the outcome in the case metadata; the outcome is only mentioned in the text of the hearing. Therefore, one needs

[5] https://github.com/chinmusique/outcome-prediction

to extract the outcomes from text manually or automatically. In this section, we outline the methodology for automatic outcome extraction, explain how sentences containing the outcome can affect subsequent outcome prediction, and delve into the details of how the annotated parts of CASELAW4 were achieved.

4.1. Manual Outcome Annotation

As the data in CAP do not contain any outcome labels, we are faced with the so-called "cold start" problem: to train a model that extracts outcomes from case reports, one needs to get some labeled data first. For this reason, we randomly selected 500 cases from the New Mexico Court of Appeals and manually annotated them with one outcome label. In total, we collected 240 AFFIRM cases (among which 12 are dismissed cases), 159 REVERSE cases, and 101 MIXED cases.

In addition to case-level labels, we annotated each case at the sentence level, identifying sentences that contain the outcome information (e.g., *Judgment is affirmed* or *We affirm in part and reverse in part*). Such outcome sentences usually appear in the summary and conclusion sections of a report, but may as well appear in the main body of the case text. Outcome sentences are needed for two reasons: on the one hand, at extraction time, pre-filtering outcome sentences leads to more accurate outcome extraction in our setting (more on it below); on the other hand, at prediction time, it is important to remove explicit mentions of the outcome from the case report, so that the results are not biased.

The annotation process was performed using the web-based annotator system from Cognitiv+ [11].[6] All annotations are made available on GitHub[7].

4.2. Outcome Extraction Methodology

We split the process of extracting the outcome from a case into two steps. First, we select all the sentences in the case report that contain the outcome description (e.g., *The chancellor's order for alimony will be continued until final decree is entered on remand of the cause. In other respects the decree will be affirmed.*), then we decide on the final outcome based on the pre-filtered sentences only (e.g., AFFIRM). The first step uses a deep learning model, while the second step uses simple keyword matching. The choice for such architecture is motivated by the following.

- We could not use a deep learning model to accurately extract outcomes from the full case report (as opposed to outcome sentences only), since there is simply not enough training data: 500 annotated cases are too few to train all the weights and parameters of a complex RNN.

- We could not perform simple keyword matching on the full texts either: since the primary purpose of outcome extraction is to label cases before outcome prediction, the labels must be sufficiently accurate, so as not to propagate the annotation error further to the predictor. As discussed in Section 4.4, the keywords and patterns need to be quite generic, so that we account for different writing styles in multiple jurisdictions, and those vary a lot. If we use a set of more specific patterns (e.g., *we accordingly affirm* or *defendant's conviction is reversed*), a lot of outcomes are omitted. While experiment-

[6] `https://cognitivplus.com/graybox`
[7] `https://github.com/chinmusique/outcome-prediction`

	Precision	Recall	F-measure
OUTCOME	97.90	95.89	96.89
NON-OUTCOME	96.49	98.21	97.35
Total	97.15	97.13	97.13

Table 3. Performance of the sentence classification model (%)

ing with sets of phrase-based patterns, we were unable to reach the F-measure higher than 81.32%, due to low recall. Conversely, as we reduced the set of patterns towards the outcome keywords *affirm*, *reverse*, *remand*, and *dismiss*, the percentage of outcomes matched by the patterns increased. However, the precision drops: keyword patterns tend to also match legal facts and reasoning, such as in *In Ark. S&L Bd. v. Grant Cty. S&L, supra, the issue was not presented and we affirmed*, or *It is contended by appellant that the judgment should be reversed*, or *Under Rule 6(c), this court shall not affirm or revert a case based on an abbreviated record*. Simple keyword matching over full reports would have inferred that the outcomes for the above examples are AFFIRM, REVERSE, and MIXED, respectively, although it is not the case.
The two-step procedure of first selecting the outcome sentences and then inferring the outcome aims to balance the precision and recall of outcome extraction, while reaching a near perfect annotation quality (see Section 4.3).

- Finally, we are unable to use complex statistical models in the second step of the extraction process for an already familiar reason: 500 annotated cases are still not enough to train an accurate deep learning model.

4.3. Sentence Classification

In order to develop a sentence-level classifier that identifies whether a given sentence contains the outcome information, we split the 500 annotated cases into individual sentences and labeled all non-outcome sentences with the NON-OUTCOME class. For example, in the following excerpt from a CAP case report, the first sentence is non-outcome, while the second sentence mentions the outcome:
*The sole question raised on appeal is whether the district court erred in determining that Defendant was subject to being sentenced as a fourth-time DWI offender instead of a third-time offender (*NON-OUTCOME*). For the reasons discussed herein, we affirm the district court's judgment and sentence (*OUTCOME*).*

In total, we got 92k sentences, including 1455 outcome sentences and 90.8k non-outcome sentences. We then re-balanced the dataset by limiting the NON-OUTCOME class to 1455 randomly selected sentences with the corresponding label; the final sentence-level dataset consisted of 2910 sentences.

We formulated the task of identifying the outcome sentences as a binary classification problem, split the sentences into training, validation, and test sets by the 8:1:1 ratio, and trained a series of bi-LSTM models. The hyperparameters were chosen from embedding size {200, 300, 2000}, input size {100, 300}, and hidden layer dimensions {50, 100, 128}. The top performing classification model is a bi-LSTM with a single hidden layer of size 50 that uses Adam optimiser [3] and has the following parameters: learning rate 0.001, embedding size 200, input size 100, and 10 epochs. It achieves an F-measure of 97.13%. Performance details of the sentence classification model are outlined in Table 3.

All experiments were implemented in PyTorch and performed on a MacBook Air laptop with macOS 10.14, 1.6 GHz Intel Core i5 processor, and 16 GB 2133 MHz LPDDR3 memory.

4.4. Outcome Extraction

Once the outcome sentences are extracted, we apply simple keyword-based patterns to identify the final outcome contained in the sentences, since it does not make sense to use data-hungry deep learning models on such a small sample of hand-annotated data. The patterns are straightforward and function as follows: if the pre-filtered sentences contain a token *affirm* or *dismiss*, the outcome is AFFIRM; if they contain a token *reverse*, the outcome is REVERSE; if both *affirm/dismiss* and *reverse* are present, the outcome is MIXED.

The above patterns prove to work extremely well, once the outcome sentences are filtered out correctly (although they are not able to work on their own, as they would not differentiate between outcomes like *Judgment affirmed on all accounts* and recitals of previous decisions of the appeal like *Judgment affirmed by the previous court ruling*; see Section 4.2). The sentence classification model is easily trained on data coming from the same jurisdiction. However, the precision and recall drop when we transfer the model to cases from other jurisdictions. Most mistakes in annotation stem from the fact that different jurisdictions use different wordings and writing styles to record the same thing. This might involve the out-of-vocabulary problem: New Mexico judges do not typically use phrases like *motion allowed* or *petition denied* to pinpoint the outcome. Errors might as well stem from grammatical variability: while in our training set, most outcomes are expressed through constructs like *we affirm/reverse* and not through *order will be affirmed/reversed*, the LSTM model did not have enough training data to generalize beyond the writing style of one jurisdiction, i.e., New Mexico. As a result, AFFIRM and REVERSE cases are not recognized, but are automatically assigned to the MIXED category, and we were not able to achieve an F-measure higher than 60% in our empirical evaluation.

Since our outcome extraction procedure is used primarily for the purpose of annotating large volumes of cases from CASELAW4, the accuracy of outcome extraction must be the highest possible, and 60% is not enough. Therefore, we augmented the sentence classification model with one additional step, also pattern-based. The idea is simple: to help deep learning generalize across jurisdictions in the absence of enough labeled data, we pair its predictions with unambiguous patterns that univocally signal the final outcome but might not have yet been captured by the model. We can easily come up with these patterns from domain expertise. If such a sentence-level pattern is matched, the sentence is labeled with the respective outcome disregarding the statistically predicted label. The sentence-level patterns that we used are: *The trial/district court's order/judgment/decision/conviction is affirmed/reversed*, *The order/judgment/decision/conviction of the district/trial court is affirmed/reversed*, *We affirm the order/judgment/decision/conviction of the district/trial court*, and *Affirmed/Reversed/Dismissed/Error/No error*.

We validated the hybrid outcome extraction model by manually checking the labels of 100 randomly selected cases, 25 per jurisdiction. The weighted average F-measure of the outcome extraction model is 82.32%. Tables 4 and 5 outline the detailed results

	Precision	Recall	F-measure
AFFIRM	93.18	78.85	85.42
REVERSE	100.00	80.77	89.36
MIXED	54.29	86.36	66.67
Total	86.40	81.00	82.32

Table 4. Performance of the outcome extraction model (%)

		Predicted label		
		AFFIRM	REVERSE	MIXED
True label	AFFIRM	41	0	11
	REVERSE	0	21	5
	MIXED	3	0	19

Table 5. Confusion matrix

Outcome type	New Mexico	Arkansas	North Carolina	Illinois	**Total**
AFFIRM	8707	33202	44022	85706	171637
REVERSE	4961	14912	16694	47933	84500
Not annotated	4658	11582	36867	49132	102239

Table 6. Number of cases per outcome type

of model validation and the confusion matrix, respectively. They demonstrate that the single most important source of errors is the AFFIRM cases that are classified as MIXED, which in turn affects the overall performance of the model. While the average precision of 86.4% would be sub-optimal for large-scale outcome extraction and case annotation, if we only focus on the AFFIRM and REVERSE classes, the weighted average precision will be 95.45%. This is the reason why our final annotations only contain AFFIRM and REVERSE, which we consider reliable.

4.5. Automated CASELAW4 *Annotation*

Finally, we used the outcome extraction model to annotate cases in CASELAW4. Since we aim for reliable, high-quality annotations, and precision is much more important than recall, we only keep the predicted labels AFFIRM and REVERSE, and we leave unlabeled the cases for which the predicted outcome is MIXED. In total, the number of labeled classes in CASELAW4 are 171637 for AFFIRM and 8450 for REVERSE; 102239 cases are left without outcome annotation. Table 6 outlines the distribution of outcome types per jurisdiction.

4.6. Lessons Learned

Outcome extraction from cases of appeal proves to be a non-trivial task. While at first glance it seems that the ways outcomes are manifested in text are quite repetitive and pattern-like (*a judgment/order/conviction/sentence is affirmed/reversed/dismissed*), there is no one straightforward way to extract outcomes automatically with high quality, for two reasons. Patterns may work well on a coherent, homogeneous set of cases, i.e., those coming from the same court. However, the language in general and the outcome sentences in particular vary a lot across courts, judges, and jurisdictions. This variability may be captured with patterns or statistical models—but for that, considerable amounts of cases from diverse sources need to be analyzed and annotated manually. The labelled data bottleneck is one of the reasons why legal outcome prediction for English language is not yet as developed as the one for Chinese language [10]. The current work aims to remedy this problem with a combination of pattern- and deep learning-based approaches, as well as to open the discussion about the value of structured legal data.

5. Summary and Outlook

This paper presents the baseline for extracting legal outcomes from the US Case Law. The main contributions of this work are the annotated dataset of English language court cases with the outcomes explicitly stated in the metadata, as well as the baseline model for outcome extraction for state Courts of Appeal cases using the Caselaw Access Project data. The new dataset CASELAW4 contains both automatic and manual annotations, and acts as the first step towards outcome prediction and advanced legal analytics for the English language legal documents, and for US state Courts of Appeal in particular.

Additionally, the work provides valuable insights into the problem of automatic annotation of legal cases. In the absence of large numbers of hand-annotated data, high-quality information extraction such as outcome extraction requires a combination of statistical learning and pattern matching. While deep learning models can typically generalize patterns appearing in texts, in the setting of labeled data deficiency, they work best when they (a) are paired with keyword- and phrase-based patterns, and (b) "mimic" keyword matching by utilizing few parameters and a small encoding size. The intuition behind it is to make them learn outcome patterns quicker. This way, the models are still versatile and are able to account for linguistic ambiguity and variability, while learning the key outcome features from little data.

The current work can be advanced in several directions. Firstly, the CASELAW4 dataset could be used in a number of prediction models, from more complex LSTMs to transformers to pre-trained language models. Since the problem of legal outcome prediction is a highly complex problem that relies on numerous factors, sophisticated deep learning models show promising results [12,19,25]. Secondly, it is important to further improve outcome extraction, to go beyond the binary system of AFFIRM and REVERSE labels and to move to more granular MIXED cases. Lastly, it is essential to further improve the outcome extraction quality by handling the linguistic variance in writing styles across courts and jurisdictions .

Acknowledgments. This work was supported by the Industrial Strategy Challenge Fund's (ISCF) Next Generation Services Research Programme and UK Research and Innovation (UKRI), through the project "Unlocking the Potential of AI for English Law".

References

[1] Hochreiter S, Schmidhuber J. Long short-term memory. Neural computation. 1997 Nov;9(8):1735-80.
[2] Wang C, Yang H, Bartz C, Meinel C. Image captioning with deep bidirectional LSTMs. In Proceedings of the 24th ACM International Conference on Multimedia 2016 Oct (pp. 988-997).
[3] Kingma DP, Ba J. Adam: A method for stochastic optimization. arXiv:1412.6980. 2014.
[4] Ashley KD. Artificial intelligence and legal analytics: New tools for law practice in the digital age. Cambridge University Press; 2017 Jul 10.
[5] Nowak J, Taspinar A, Scherer R. LSTM recurrent neural networks for short text and sentiment classification. In Proceedings of the 19th International Conference on Artificial Intelligence and Soft Computing 2017 Jun (pp. 553-562).
[6] Liang Q, Wu P, Huang C. An efficient method for text classification task. In Proceedings of the 2019 International Conference on Big Data Engineering 2019 Jun 11 (pp. 92-97).
[7] Lim CG, Choi HJ. LSTM-based model for extracting temporal relations from Korean text. In Proceedings of the 2018 IEEE International Conference on Big Data and Smart Computing 2018 (pp. 666-668).
[8] The President and Fellows of Harvard University. Caselaw Access Project. 2018, `https://case.law`.

[9] Spaeth HJ, Epstein L, Martin AD, Segal JA, Ruger TJ, Benesh SC. 2016 Supreme Court Database, Version 2016 Legacy Release v01. (SCDB_Legacy_01) http://supremecourtdatabase.org.
[10] Xiao C, Zhong H, Guo Z, Tu C, Liu Z, Sun M, Feng Y, Han X, Hu Z, Wang H, Xu J. CAIL2018: A large-scale legal dataset for judgment prediction. arXiv preprint arXiv:1807.02478. 2018 Jul 4.
[11] Cognitiv+ blog. Cognitiv+ is offering our NLP annotator free of charge to all Covid-19 researchers. 2020, https://cognitivplus.com/cognitiv-offers-nlp-annotator-free-of-charge-to-all-research-projects-fighting-to-find-the-cure-against-covid-19.
[12] Chalkidis I, Androutsopoulos I, Aletras N. Neural legal judgment prediction in English. In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, 2019 Jul (pp. 4317-4323).
[13] Aletras N, Tsarapatsanis D, Preoţiuc-Pietro D, Lampos V. Predicting judicial decisions of the European Court of Human Rights: A natural language processing perspective. PeerJ Computer Science. 2016.
[14] Chalkidis I, Fergadiotis E, Malakasiotis P, Androutsopoulos I. Large-scale multi-label text classification on EU Legislation. In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, 2019 Jul (pp. 6314-6322).
[15] Medvedeva M, Vols M, Wieling M. Using machine learning to predict decisions of the European Court of Human Rights. Artificial Intelligence and Law. 2020 Jun;28(2):237-66.
[16] Medvedeva M, Vols M, Wieling M. Judicial decisions of the European Court of Human Rights: Looking into the crystal ball. In Proceedings of the Conference on Empirical Legal Studies 2018.
[17] Katz DM, Bommarito MJ, Blackman J. A general approach for predicting the behavior of the Supreme Court of the United States. PloS one. 2017 Apr;12(4):e0174698.
[18] Şulea OM, Zampieri M, Vela M, van Genabith J. Predicting the law area and decisions of French Supreme Court cases. In Proceedings of the International Conference Recent Advances in Natural Language Processing, 2017 Sep (pp. 716-722).
[19] Lim C. An evaluation of machine learning approaches to Natural Language Processing for legal text classification [master thesis]. Imperial College London; 2019.
[20] Zhong H, Guo Z, Tu C, Xiao C, Liu Z, Sun M. Legal judgment prediction via topological learning. In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing 2018 (pp. 3540-3549).
[21] Yang W, Jia W, Zhou X, Luo Y. Legal judgment prediction via multi-perspective bi-feedback network. In Proceedings of the 28th International Joint Conference on Artificial Intelligence 2019 Aug 10 (pp. 4085-4091).
[22] Luo B, Feng Y, Xu J, Zhang X, Zhao D. Learning to predict charges for criminal cases with legal basis. In Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing 2017 Sep (pp. 2727-2736).
[23] Ye H, Jiang X, Luo Z, Chao W. Interpretable charge predictions for criminal cases: Learning to generate court views from fact descriptions. In Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 2018 Jun (pp. 1854-1864).
[24] Hu Z, Li X, Tu C, Liu Z, Sun M. Few-shot charge prediction with discriminative legal attributes. In Proceedings of the 27th International Conference on Computational Linguistics 2018 (pp. 487-498).
[25] Chen H, Cai D, Dai W, Dai Z, Ding Y. Charge-based prison term prediction with deep gating network. In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing 2019 Nov (pp. 6363-6368).
[26] Li C, Sheng Y, Ge J, Luo B. Apply event extraction techniques to the judicial field. In Adjunct Proceedings of the 2019 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the 2019 ACM International Symposium on Wearable Computers 2019 Sep 9 (pp. 492-497).
[27] Šavelka J, Ashley KD. Extracting case law sentences for argumentation about the meaning of statutory terms. In Proceedings of the Third Workshop on Argument Mining 2016 Aug (pp. 50-59).
[28] Copus R, Hübert R. Detecting Inconsistency in Governance. Working Paper, available at https://papers.ssrn.com/sol3/papers.cfm?abstract_id=2812914.
[29] Vacek T, Teo R, Song D, Nugent T, Cowling C, Schilder F. Litigation Analytics: Case outcomes extracted from US federal court dockets. In Proceedings of the Natural Legal Language Processing Workshop 2019 Jun 7 (pp. 45-54).
[30] Li Q, Zhang Q, Yao J, Zhang Y. Event extraction for criminal legal text. In Proceedings of the 2020 IEEE International Conference on Knowledge Graph 2020 Aug (pp. 573-580).