Legal Information Retrieval

ECIR 2023 workshop proceedings

April 2nd, 2023

Abstract. Although this is the first legal IR workshop organized at ECIR, the topic has a long history of prior successful events and benchmark campaigns. The workshop covers a broad variety of tasks, challenges, and methods in the legal domain. We have three invited speakers, oral and poster presentations based on extended abstracts. The workshop schedule is available at https://tmr.liacs.nl/legalIR/

Introduction

Legal professionals spend up to a third of their time doing research and investigation. Two specific legal tasks that have attracted the attention of the Information Retrieval (IR) community in the past decades are eDiscovery and case law retrieval. Both are tasks that are strongly recall-oriented. Other legal IR tasks have received less attention, for example legal web search in commercial legal search engines, legal community question answering, and lawyer finding. In this workshop, we address the complete scope of legal IR tasks, challenges, and methods needed to address those challenges.

The LegalIR workshop is a full-day workshop with talks by invited speakers, talks and posters based on submitted extended abstracts, and discussion.

Workshop contents

Invited speakers

Maura R. Grossman (Research Professor in the School of Computer Science at the University of Waterloo, an Adjunct Professor at Osgoode Hall Law School of York University, and an affiliate faculty member at the Vector Institute of Artificial Intelligence): **The Limitations and Misuse of Information Retrieval in Legal Cases**

The authors recently undertook extensive validation testing on review technologies and processes to be employed in relation to a large, high-value/high-stakes legal case in Ireland. It was understood that legal discovery in these proceedings, with an estimated data set of over a quarter-billion documents, was not likely to be viable using commercially available electronic discovery software or conventional document review methods. Thus, as an alternative, a CAL® review platform, developed by Maura R. Grossman and Gordon V. Cormack was to be used to provide the active learning functionality underpinning the review.

This CAL® system was designed to incorporate state-of-the-art developments in active learning as applied to legal information retrieval. Review practices, informed by the results of academic research into the limitations of human review, were to be used, both in parallel with and as an alternative to methods more typically employed in current legal practice. We report on the results of this research.

Milda Norkute (Lead Designer at Thomson Reuters Labs in Zug, Switzerland): **Evaluation of legal search from the end users' perspective – current challenges and opportunities in industry**

Legal research is ambiguous, challenging and time-consuming. This is because the "answers" to legal research questions are often not found in a single document and finding the answer can require putting together non-obvious and sometimes contradictory information from multiple documents and different sources. Therefore, designing and building search for legal professionals has unique challenges. This talk will focus on the question: how might we evaluate user's search experience by using cross-disciplinary methods and inform the development of next generation of search for legal professionals? Existing practices to empathise with the users and understand their experience as well as questions that still remain unanswered will be discussed.

Tjerk de Greef (Director of Advanced & Search Technology in the global technology organization of Wolters Kluwer): **Actionable content – how semantic data boosts legal professional search**

Today, Wolters Kluwer focusses on creating top-notch online services for a variety of professional customers worldwide. Everything we do is driven by state-of-the-art software, including Machine Learning based content enrichment microservices with the goal to enable advanced, complete, and semantic search experience for the legal professionals we support. A central pillar in these 'expert solutions' is actionable content, with the goal to transform and/or enriched existing (public) content sources. Actionable content allows customers to leverage the knowledge in these documents and align it in a way that is integrated into their daily work. Such approaches require a pivotal thinking. The technology stack moves away from searching documents to true semantic search. In other words: the goal is to leverage data points that have a semantic meaning and created a linked graph. In this talk, we will elaborate on the NLP toolbox that enabling a better understanding of documents, including addressing endeavors in support of Legal Analytics and supporting asking question in natural legal language. We will also address our Machine Learning Life Cycle and toolbox to validate and measure search quality. Lastly, I will also address the approach Wolters Kluwer is following centralize around UX design patterns and link actionable content.

Presentations based on extended abstracts

- Maren Pielka, David Biesner, Rajkumar Ramamurthy, Tim Dilmaghani Khameneh, Bernd Kliem, Rüdiger Loitz and Rafet Sifa: Improving Automated Auditing Systems with Zero-Shot Text Matching and Sentence Transformers

- Masaharu Yoshioka, Juliano Rabelo, Randy Goebel, Yoshinobu Kano, Mi-Young Kim and Ken Satoh: Competition on Legal Information Extraction/Entailment (COLIEE)
- Aimen Louafi and Pauline Chavallard: Finding Unstructured References to French Collective Agreements in Legal Documents
- Alexandre G. Lima, Jose G Moreno, Mohand Boughanem, Taoufiq Dkaki and Eduardo Aranha: Leveraging Positional Encoding to Improve Fact Identification in Legal Documents
- Adam Wyner, Adeline Nazarenko, François Lévy and Haifa Zargayouna: Semantic Search in Legislation
- Tobias Fink, Yasin Ghafourian, Georgios Peikos, Florina Piroi and Allan Hanbury: An Annotation Framework for Benchmark Creation in the Legal Case Retrieval Domain
- Nishchal Prasad, Mohand Boughanem and Taoufiq Dkaki: Exploring Semi-supervised Hierarchical Stacked Encoder for Legal Judgement Prediction
- Behrooz Mansouri and Ricardo Campos: FALQU: Finding Answers to Legal Questions
- Charles Courchaine and Ricky Sethi: Opening the TAR Black Box: Developing an Interpretable System for eDiscovery Using the Fuzzy ARTMAP Neural Network
- Kees van Noortwijk and Christian Hirche: Parsing User Queries using Context Free Grammars
- David Lewis: Implicit Assumptions in the Evaluation of One-Phase Technology-Assisted Review

Organizing committee

- Suzan Verberne, Leiden Institute of Advanced Computer Science, Leiden University, The Netherlands
- Evangelos Kanoulas, Informatics Institute, University of Amsterdam, The Netherlands
- Gineke Wiggers, eLaw Center for Law and Digital Technologies, Leiden University, The Netherlands
- Florina Piroi, Institute of Information Systems Engineering, TU Wien, Austria
- Arjen P. de Vries, Institute for Computing and Information Sciences, Radboud University, The Netherlands

In addition to the organizers acting as reviewers we had four external reviewers, who we thank for their work: Daniel Locke, Gábor Recski, Julien Rossi, Procheta Sen.

Proceedings

All extended abstracts that are presented in the workshop are following on the next pages.

Improving Automated Auditing Systems with Zero-Shot Text Matching and Sentence Transformers

Maren Pielka[†], David Biesner[†‡], Rajkumar Ramamurthy[†‡], Tim Dilmaghani Khameneh[§], Bernd Kliem[§], Rüdiger Loitz[§], Rafet Sifa[†]
[†] Fraunhofer IAIS (name.surname@iais.fraunhofer.de), [‡] University of Bonn,
[§] PwC GmbH WPG (name.surname@pwc.com)

ABSTRACT

In this work, we study the efficiency of unsupervised text matching using Sentence-Bert, a transformer-based model, by applying it to semantic similarity matching of text paragraphs in financial reports. Experimental results show that this model is robust to documents from in- and out-of-domain data.[1]

KEYWORDS

NLP, Transfer Learning, BERT, Text Classification

ACM Reference Format:
Maren Pielka[†], David Biesner[†‡], Rajkumar Ramamurthy[†‡], Tim Dilmaghani Khameneh[§], Bernd Kliem[§], Rüdiger Loitz[§], Rafet Sifa[†]. 2023. Improving Automated Auditing Systems with Zero-Shot Text Matching and Sentence Transformers. In *Proceedings of Legal Information Retrieval Workshop at the 45th European Conference on Information Retrieval (ECIR) (ECIR LIR '23).* ACM, New York, NY, USA, 2 pages. https://doi.org/XXXXXXX.XXXXXXX

1 INTRODUCTION

The auditing of financial reports is a costly and time-consuming task, which has to be performed manually by trained auditors. They have to determine whether the report has been prepared according to all legal requirements of the applicable financial reporting framework (e.g. the International Financial Reporting Standards, IFRS). In order to accomplish that, the auditor has to find the relevant section for each item of the framework checklist in the document. This step can be facilitated considerably by using a machine learning based recommendation system, which would show to the user the most relevant text passages from the document for each requirement. This has been successfully addressed by us in previous work [4, 6], where the Automated List Inspector (ALI) as a tool for semi-automated auditing was presented. It is being used in practice by PriceWaterhouse Coopers since 2019.

In this work, we present an enhancement of ALI, namely the matching of text passages and requirements independently of any specific underlying checklist, which minimizes the need for retraining after changes to the checklist. Based on textual semantic similarity, our proposed model encodes both the requirement and paragraph text to latent representations and calculates the cosine similarity between the two vectors to get a relevance score. Such a model is then capable of matching texts to entirely new requirements it has not seen during training.

2 METHODOLOGY

Our proposed method for text matching is based on the Sentence-BERT architecture [5], which is itself building upon the BERT [2] framework. The SentenceBERT model encodes both input texts via two BERT models with shared model weights. To convert the token embeddings to sentence embeddings, we apply a mean-pooling layer. The cosine similarity between the text embeddings is then calculated as a measure for how well the two input texts match. To predict and recommend checklist items for a paragraph, the paragraph and all checklist item texts are encoded by the model. The cosine similarity between the paragraph and each checklist item is calculated and sorted. The model outputs the top-k checklist items with the highest similarity score.

3 DATA

For unsupervised training, we provide two datasets of German and English language. The German dataset consists of financial reports from BundesAnzeiger[2] (*BANZ*). The English dataset consists of financial reports from the US Securities and Exchange Commission[3] (*SEC*). For supervised training, we annotate two datasets of financial reports in German and English language. Those data sets were provided to us by an auditing firm, who collaborated with us for this project. We annotate each paragraph as, if applicable, corresponding to one or more individual checklist items from the IFRS regulatory checklist with a total of 1305 items. For model testing, we create two test splits. One test set stems from the same distribution as the training set, and contains only requirement annotations for the same requirements as the annotations in the training set. We call these test sets *test seen*, as the model has already seen the respective requirements during training. The other type of test set contains new requirement annotations that the model has not seen during training, i.e. no report text in the training dataset has any annotation of these requirements. We call these test sets *test unseen*. Both test sets contain only text passages from reports that were not observed during training.

[1] This work was previously published in proceedings of IEEE International Conference on Machine Learning Applications IEEE ICMLA 2022 [1].

ECIR LIR '23, April 02, 2023, Dublin, Ireland

ACM ISBN 978-1-4503-XXXX-X/18/06…$15.00
https://doi.org/XXXXXXX.XXXXXXX

[2] https://www.bundesanzeiger.de/
[3] https://www.sec.gov/dera/data/financial-statement-and-notes-data-set.html

4 TRAINING DETAILS

We pre-train the *bert-base-multilingual-cased* language model further on a dataset of financial language data (*BANZ* and *SEC*) using masked language modeling.

The second training step consists of training the sentence embeddings in an unsupervised manner. During this training stage, we do not differentiate between report texts and requirements and consider both as individual input texts. We consider two training mechanisms: Simple Contrastive Learning of Sentence Embeddings (*SimCSE* [3]) and Tranformer-based Denoising AutoEncoder (*TSDAE* [7]). For details on the training methods, we refer to the respective papers. We train the model using both methods on two unannotated datasets of German (*BANZ*) and English (*SEC*) language until convergence, and select the model with the best matching score on the validation set for further training or evaluation.

During the last training step, we aim to utilize the annotated datasets, i.e. the datasets of financial reports in German and English with report paragraphs annotated as matching a certain requirement text. The training procedure is based on contrastive learning, similar to the SimCSE. We train the model to maximize the cosine similarity of matching text and requirement pairs and minimize the cosine similarity of all other pairs. We train the model on German, English or German and English annotated reports (*DE*, *EN* and *DE+EN* in Table 1) until convergence and select the model with the best matching score on the validation set for evaluation.

5 EXPERIMENTS AND RESULTS

We collect the evaluation results in Table 1. For space reasons, we present partial results here; for the full evaluation, please refer to our original paper [1]. We compute the one-shot recall on the top-5 recommendations given by the model, which is a custom performance measure that has been designed to reflect user experience. We assign a score of 1.0 to a sample if a correct requirement is part of the model's top 5 predictions for a given text passage, and average the score over the dataset.

		DE		EN		DE+EN	
Unsupervised Method	Supervised Training	Seen	Unseen	Seen	Unseen	Seen	Unseen
Language Modeling Only		0.7	0.0	0.6	0.1	0.7	0.7
TSDAE	-	12.3	0.1	17.8	0.5	14.5	0.8
SIMCSE	-	13.4	18.1	16.4	0.4	14.6	0.8
TSDAE	DE	**91.6**	33.9	52.4	**56.3**	76.0	**47.4**
TSDAE	EN	56.1	44.1	86.1	26.8	68.1	33.6
TSDAE	DE+EN	88.6	46.3	**92.8**	45.6	**90.3**	45.9
SIMCSE	DE	88.4	34.5	50.8	45.2	73.4	41.0
SIMCSE	EN	51.6	30.5	87.7	30.1	66.0	30.3
SIMCSE	DE+EN	83.7	**47.5**	84.4	46.7	84.0	47.0

Table 1: Evaluation of all training runs on the hold-out test datasets. We compare the models trained purely as language models, trained unsupervised using the TSDAE and SIMCSE method, and further trained in a supervised manner.

We first note that only unsupervised training results in poor performance on the test sets, with a maximum of 17.81% one-shot recall on a *seen* test set. We therefore concentrate on the effect of the unsupervised training method when continuing training in a supervised manner, see the bottom half of Table 1. We see that the best TSDAE model outperforms the best SimCSE model by a significant margin on all *seen* test sets (improvement of 3 to 5 percentage points) and two of the three *unseen* test sets (improvement of 7 to 10 percentage points). For the remaining *unseen* test set the metric scores of the best models are very similar (46.32% for TSDAE and 47.45% for SimCSE). As a result of this evaluation, we conclude that TSDAE is the more fitting unsupervised training method.

Considering the effect of supervised training data language on the test performance, we see no significant trend towards any specific data language setup. While the model trained on German language only performs best on the German *seen* test set, the model trained on English language only does not outperform the other data setups in any test set. But training on any English language data is clearly necessary for the *seen* test sets in English or combined English and German.

In general, the evaluation shows that the proposed model architecture performs very well on the task of predicting requirements available in training data (over 90% one-shot recall on all *seen* datasets). Additionally, the methods produce reasonable results when faced with new requirements, which a multilabel prediction model could not process at all.

6 CONCLUSION

We found that the proposed method performs competitively on *seen* labels and reasonably well on *unseen* labels. This suggests these models to be useful in a setting were a subset of the labels or the entire label set might change between retrainings of a model. Future work includes application of the model to unseen languages, combining both unsupervised pretraining methods, and using separate encoders for requirement and report text.

ACKNOWLEDGMENTS

In parts, this research has been funded by the Federal Ministry of Education and Research of Germany and the state of North-Rhine Westphalia as part of the Lamarr-Institute for Machine Learning and Artificial Intelligence, LAMARR22B.

REFERENCES

[1] David Biesner, Maren Pielka, Rajkumar Ramamurthy, et al. 2022. Zero-Shot Text Matching for Automated Auditing using Sentence Transformers. In *Proc. ICML-A.*
[2] J. Devlin, Ming-Wei Chang, et al. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proc. NAACL-HLT.* https://doi.org/10.18653/v1/N19-1423
[3] Tianyu Gao et al. 2021. SimCSE: Simple Contrastive Learning of Sentence Embeddings. https://doi.org/10.48550/ARXIV.2104.08821
[4] Rajkumar Ramamurthy, Maren Pielka, et al. 2021. ALiBERT: Improved Automated List Inspection (ALI) with BERT. In *Proceedings of the 21st ACM Symposium on Document Engineering* (Limerick, Ireland) *(DocEng '21).* Association for Computing Machinery, New York, NY, USA, Article 20, 4 pages. https://doi.org/10.1145/3469096.3474928
[5] Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing.* Association for Computational Linguistics. https://arxiv.org/abs/1908.10084
[6] Rafet Sifa, Anna Ladi, et al. 2019. Towards Automated Auditing with Machine Learning. In *Proceedings of the ACM Symposium on Document Engineering 2019* (Berlin, Germany) *(DocEng '19).* Association for Computing Machinery, New York, NY, USA, Article 41, 4 pages. https://doi.org/10.1145/3342558.3345421
[7] Kexin Wang et al. 2021. TSDAE: Using Transformer-based Sequential Denoising Auto-Encoder for Unsupervised Sentence Embedding Learning. https://doi.org/10.48550/ARXIV.2104.06979

Competition on Legal Information Extraction/Entailment (COLIEE)

Masaharu Yoshioka*
yoshioka@ist.hokudai.ac.jp
Faculty of Information Science and Technology, Hokkaido University
Sapporo-shi, Hokkaido, Japan

Juliano Rabelo
Randy Goebel
rabelo@ualberta.ca
rgoebel@ualberta.ca
Alberta Machine Intelligence Institute, University of Alberta
Edmonton, Alberta, Canada

Yoshinobu Kano
kano@inf.shizuoka.ac.jp
Faculty of Informatics, Shizuoka University
Hamamatsu, Shizuoka, Japan

Mi-Young Kim
miyoung2@ualberta.ca
Dept. of Science, Augustana Faculty, University of Alberta
Camrose, Alberta, Canada

Ken Satoh
ksatoh@nii.ac.jp
National Institute of Informatics
Chiyoda-ku, Tokyo, Japan

CCS CONCEPTS

• **Information systems → Information extraction**; **Specialized information retrieval**.

KEYWORDS

legal information, information retrieval, entailment, datasets

ACM Reference Format:
Masaharu Yoshioka, Juliano Rabelo, Randy Goebel, Yoshinobu Kano, Mi-Young Kim, and Ken Satoh. 2023. Competition on Legal Information Extraction/Entailment (COLIEE). In *Proceedings of The first international workshop on Legal Information Retrieval (LegalIR '23).* ACM, New York, NY, USA, 2 pages. https://doi.org/XXXXXXX.XXXXXXX

1 INTRODUCTION

The Competition on Legal Information Extraction/Entailment (COLIEE) [1] is a series of competition workshops for Legal Information Extraction/Entailment that started from 2014 (original idea was introduced in [4]) and is currently held yearly[3, 5–8]. There are two types of tasks (Statute law task and Case law task) characterized by the legal documents used for each component of the competition. Here we briefly introduce these tasks and summarize the methods used over the history of this competition.

*All authors contributed equally to this research.
[1]https://sites.ualberta.ca/~rabelo/COLIEE2022/

LegalIR '23, April 2, 2023, Dublin, Ireland

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM…$15.00
https://doi.org/XXXXXXX.XXXXXXX

```
<pair id="R02-9-E" label="N">
<t1>Article 192 A person that commences the
possession of movables peacefully and openly
by a transactional act acquires the rights that
are exercised with respect to the movables
immediately if the person possesses it in good
faith and without negligence.</t1>
<t2>B obtained A's bicycle by fraud. In this
case, A may demand the return of
the bicycle against B by filing an action for
recovery of possession.</t2></pair>
```

Figure 1: Example of training data for the Bar exam question with a relevant article and entailment label

2 TASK DETAILS

2.1 Statute law task

The statute law task is the original task of COLIEE. The target task is to check whether a question statement extracted from the Japanese Bar exam is correct or not. There are two sub-tasks in this task. One is an information retrieval (IR) task for retrieving relevant articles for a given question. The other is an entailment task for checking whether the relevant articles entail the question statement or not.

Original data is provided in Japanese, but we also provide an English translated version for the English participants. Figure 1 is an example of the data that shows a question (<t2>) and a relevant article (<t1>) with an entailment label ("N")(No) in English. Questions are collected from the Japanese Bar exam and relevant articles are defined by experts from the legal domain. Entailment results were collected from the solutions in the official Japanese Bar exam publication. The IR task uses the <t2> part without <t1> as a query and participants' competition systems return an article number list that is relevant to the query. The legal entailment task uses <t1> and <t2> as a query and participants' competition systems should return an answer "Y" (entail) or "N" for the pair of <t1> and <t2>, according to whether <t2> is entailed by <t1> or not.

For the IR task, BM25 [11] provides a strong baseline. BM25 shows a good performance when there is an exact match of legal terms between the question and articles. Recent deep learning transformer technologies such as BERT [2], including domain-specific transformers [1] are also used in the task. However, while these systems are good at the questions that require context, they tend to make mistakes when questions require an exact match of legal terms.

The main evaluation measure of the IR task is macro average of F2-measure (which places more emphasis on recall than precision), because the IR task is used as a preprocess of the entailment task and the testing of the entailment without relevant articles is almost meaningless. The system that showed the best performance in the recent COLIEE 2022 [7] uses a BM25 based system that considers the structure of the statement (condition and decision) and whose F2 score is 0.82. The evaluation measure of the entailment task is the accuracy. The system that showed the best performance in the recent COLIEE 2022 [7] uses a combination of a rule-base approach and BERT with data augmentation and achieved a score of 0.68.

2.2 Case law task

The other major task is a case law task which began in 2018 [3]. The target task is for retrieving support cases that are "noticed" in the given case and which identify the most relevant part that entails the portion of text identified in the given case. There are two sub-tasks in this task. One is an information retrieval (IR) task that retrieves support cases that are noticed in the given case. The other is an entailment task that identifies appropriate paragraphs from the referred case, which supports entailment of the paragraph in the given case.

Competition data are extracted from the Federal Court of Canada case law. Since it is difficult to provide a whole set of cases, the IR task uses a selected case database by selecting noticed cases as relevant cases and random sampled cases as the non-relevant cases. For the non-relevant cases, two legal experts check the cases as non-relevant. The IR task uses a given case as a query and participants return noticed cases from the case database. The Entailment task is defined as a passage retrieval task identified by the paragraph numbers. This task uses a part of the paragraph that noticed the target case as an input and participants return paragraph numbers of the target case for supporting the decision.

For both tasks, BM25 [11] and deep learning approaches are used. However, compared to the statute law cases, context information is more important in comparing the cases.

The main evaluation measure for these tasks is the micro-average of FF-measure. The best performance system of the latest COLIEE 2022 [7] of the IR task uses Sentence-BERT [10] for generating a distributed representation vector of each case and compare the vectors for calculating the similarity between the cases. They also apply preprocessing and postprocessing steps to decide which cases are noticed. The performing run achieved a score of 0.37. For the entailment task, the best performance system uses T5 [9] for calculating the similarity between the query and paragraphs of the target article. These competitors trained T5 with a variety of settings and combined the results to obtain the final answer. The best performance result is 0.68.

3 SUMMARY

We have briefly explained two tasks of the COLIEE (Statute-law and Case-law) with sub-task definitions, datasets used for the tasks and major approaches used in the COLIEE competition. Our effort continues and we will plan to have COLIEE 2023 as a workshop of the 19th International Conference on Artificial Intelligence and Law - ICAIL 2023.

ACKNOWLEDGMENTS

This competition would not be possible without the significant support of Colin Lachance from vLex, Compass Law and Jurisage, and the guidance of Jimoh Ovbiagele of Ross Intelligence and Young-Yik Rhim of Intellicon. Our work to create and run the COLIEE competition is also supported by our institutions: the National Institute of Informatics (NII), Shizuoka University and Hokkaido University in Japan, and the University of Alberta and the Alberta Machine Intelligence Institute in Canada. We also acknowledge the support of the Natural Sciences and Engineering Research Council of Canada (NSERC), [including DGECR-2022-00369, RGPIN-2022-0346]. This work was also supported by JSPS KAKENHI Grant Numbers, JP17H06103 and JP19H05470 and JST, AIP Trilateral AI Research, Grant Number JPMJCR20G4.

REFERENCES

[1] Ilias Chalkidis, Manos Fergadiotis, Prodromos Malakasiotis, Nikolaos Aletras, and Ion Androutsopoulos. 2020. LEGAL-BERT: The Muppets straight out of Law School. In *Findings of the Association for Computational Linguistics: EMNLP 2020*. Association for Computational Linguistics, Online, 2898–2904. https://doi.org/10.18653/v1/2020.findings-emnlp.261

[2] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *CoRR* abs/1810.04805 (2018). arXiv:1810.04805

[3] Yoshinobu Kano, Mi-Young Kim, Masaharu Yoshioka, Yao Lu, Juliano Rabelo, Naoki Kiyota, Randy Goebel, and Ken Satoh. 2019. COLIEE-2018: Evaluation of the Competition on Legal Information Extraction and Entailment. In *New Frontiers in Artificial Intelligence*, Kazuhiro Kojima, Maki Sakamoto, Koji Mineshima, and Ken Satoh (Eds.). Springer International Publishing, Cham, 177–192.

[4] Mi-Young Kim, Ying Xu, Randy Goebel, and Ken Satoh. 2014. Answering Yes/No Questions in Legal Bar Exams. In *New Frontiers in Artificial Intelligence*, Yukiko Nakano, Ken Satoh, and Daisuke Bekki (Eds.). Springer International Publishing, Cham, 199–213.

[5] Juliano Rabelo, Mi-Young Kim, Randy Goebel, Masaharu Yoshioka, Yoshinobu Kano, and Ken Satoh. 2020. A Summary of the COLIEE 2019 Competition. In *New Frontiers in Artificial Intelligence*, Maki Sakamoto, Naoaki Okazaki, Koji Mineshima, and Ken Satoh (Eds.). Springer International Publishing, Cham, 34–49.

[6] Juliano Rabelo, Mi-Young Kim, Randy Goebel, Masaharu Yoshioka, Yoshinobu Kano, and Ken Satoh. 2021. COLIEE 2020: Methods for Legal Document Retrieval and Entailment. In *New Frontiers in Artificial Intelligence. JSAI-isAI 2020. Lecture Notes in Computer Science*. Springer International Publishing, Cham, 196–210.

[7] Juliano Rabelo, Mi-Young Kim, Randy Goebel, Masaharu Yoshioka, Yoshinobu Kano, and Ken Satoh. 2022. COLIEE 2022 Summary: Methods for Legal Document Retrieval and Entailment. In *Proceedings of the Sixteenth International Workshop on Juris-informatics (JURISIN)*. 3–16.

[8] Juliano Rabelo, Mi-Young Kim, Randy Goebel, Masaharu Yoshioka, Yoshinobu Kano, and Ken Satoh. 2022. Overview and Discussion of the Competition on Legal Information Extraction/Entailment (COLIEE) 2021. 16 (04 2022), 111–133. https://doi.org/10.1007/s12626-022-00105-z

[9] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2019. Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer. arXiv:1910.10683 [cs.LG]

[10] Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics. https://arxiv.org/abs/1908.10084

[11] S. E. Robertson, S. Walker. 2000. Okapi/Keenbow at TREC–8. In *Proceedings of TREC-8*. 151–162.

Finding Unstructured References to Collective Agreements in French Legal Documents

Aïmen LOUAFI*
aimen.louafi@doctrine.fr
Doctrine
Paris, FRANCE

Pauline CHAVALLARD*
pauline@doctrine.fr
Doctrine
Paris, FRANCE

ABSTRACT

Collective agreements are documents that specify the terms and conditions of employment within a particular industry, such as the restaurant industry.

The task of detecting these agreements references in legal documents involves using text analysis and natural language processing techniques to identify and extract the agreements. The goal is to efficiently retrieve this information and link it to other relevant legal content, which can be useful for legal research and document analysis.

In this paper, an annotation scheme for this task is proposed and a dataset is built using it. An approach using dependency parsing for entity linking and embeddings for disambiguation is also proposed, achieving an F1-score of 0.82.

KEYWORDS

dataset creation, named entity recognition, entity linking, entity disambiguation, sentence embeddings, text processing for legal IR

ACM Reference Format:
Aïmen LOUAFI and Pauline CHAVALLARD. 2023. Finding Unstructured References to Collective Agreements in French Legal Documents. In *Proceedings of ACM Conference (Conference'17).* ACM, New York, NY, USA, 3 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

France has over 650 collective agreements that specify the conditions of employment, each with a unique name. Extracting these names from unstructured and very long documents (1700 words on average), such as court decisions or laws, can be difficult due to their varying length, use of commas and acronyms. Moreover, collective agreements can be cited in multiple ways:

- base text (e.g. "convention collective des bureaux d'études techniques, des cabinets d'ingénieurs-conseils et des sociétés de conseils du 15 décembre 1987")
- attached text (attached to a collective agreement, see Figure 1 for more details) (e.g. "l'annexe n°2 du 12 juin 1999 de la convention",
- a specific article of a text (e.g. "article 3-2 de la convention").

Conference'17, July 2017, Washington, DC, USA

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM…$15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

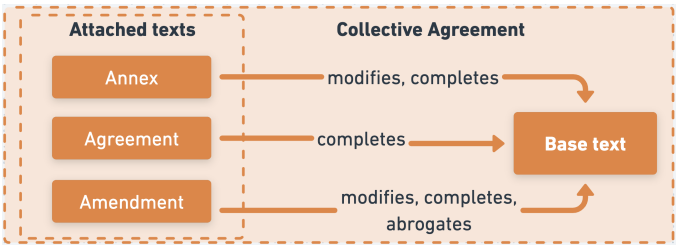


Figure 1: Collective agreement and attached texts

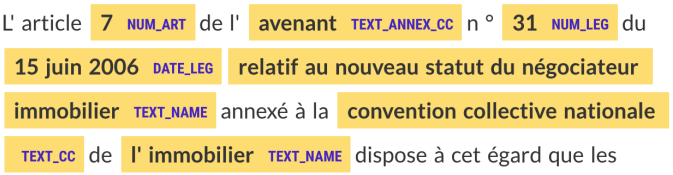


Figure 2: Annotation scheme

Additionally, agreements may be mentioned implicitly (e.g. "article 3 of the applicable collective agreement"), and some agreements may have multiple articles with the same numbering (each chapter of the text has its own numbering).

2 ANNOTATION AND DATASET BUILDING

To the best of our knowledge, there is currently no annotated dataset available for this task. Therefore, we had to create one. The initial step involved using Named Entity Recognition. The annotation process can be challenging, as it involves dealing with implicit references, article numbers that may be distant from their corresponding text, and collective agreement names that can be lengthy or mentioned before or after they are first introduced.

Figure 2 shows an example of the annotation scheme we decided to use. Collective agreements are identified using the TEXT_CC tag, and their name using the TEXT_NAME tag.

Annotation was done manually but paragraphs were pre-annotated using a regular expression baseline. The data comes directly from the French institutional site Légifrance (or DILA) in charge of publishing legislative documents. We annotated a total of approximately 4050 paragraphs, which were taken from court decisions, commentaries, and law articles.However, we found that some collective agreements were rarely cited. In order to make our dataset as exhaustive as possible, we added additional paragraphs, by artificially replacing some collective agreement names with others from a list of all collective agreement names. For example, we might

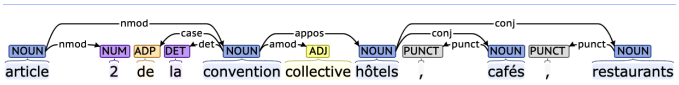


Figure 3: Dependency parsing

replace "convention collective de l'immobilier" with "convention collective de transport". We did this until we reached a total of 6000 paragraphs for our dataset.

Sometimes, when replacing a feminine noun with a masculine one, it can result in grammatically incorrect sentences (for example, "convention collective de la cinéma"). To our knowledge, there are various methods for augmenting data while preserving the linguistic coherence of generated sentences. [3] However, in our case, this was not a significant issue, as it could force the model to learn from context instead.

3 ENTITY DETECTION

First step is to detect all of the entities present in the text. Using regular expressions yields a poor F1-score of 0.1. This is because collective agreements name are lengthy, contain commas, and have multiple variations (such as "medical industry" and "medicine") etc… We decided to use a *bi-LSTM CRF* implementation, akin to the architecture used in the *Lample* paper [2], that we optimized using grid-search. This architecture works fine for this task because it handles capital letters (often the case for collective agreements names), and also uses a character based approach (handles acronyms).

4 ENTITY LINKING

Once the individual entities are extracted, it is necessary to link them together (for example, linking names to the corresponding collective agreements and connecting article numbers with the appropriate text). This is a dependency parsing task, for which we decided to use pre-existing French syntactic parsing modules. However, as none of these modules were specifically trained on legal data, we evaluated various tools. We ultimately chose the *stanza* implementation, along with manual rules to correct common errors and handle simple cases.

Figure 3 shows an example of a dependency parsing tree we get using stanza. We then use it to link the entities together with their closest candidate in the tree.

5 ENTITY MATCHING

With the entities linked, the next step is to identify the correct collective agreement that is being mentioned. Here are the details of the identification strategies in the most common cases:

CASE 1: if a collective agreement is explicitly mentioned (TEXT_CC with TEXT_NAME), we use the BM25 score between the TEXT_NAME with the set of available collective agreement names, applying stemming on the names. If the score is above a certain threshold, we return the correct collective agreement.

CASE 2: if a collective agreement is implicitly mentioned (TEXT_CC without TEXT_NAME), we retrieve the last mention to a collective agreement before this one, and use its TEXT_NAME as the TEXT_NAME of the current entity. Then, we apply CASE 1

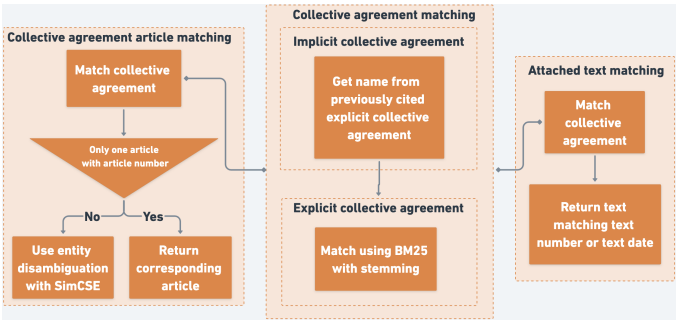


Figure 4: Entity matching

CASE 3: if a collective agreement is mentioned with an article (TEXT_CC with NUM_ART), we retrieve the right collective agreement via CASE 1 or CASE 2. If there is only one article number corresponding to the NUM_ART, we return this article. Otherwise we apply the disambiguation strategy detailed in the next section.

CASE 4: If an attached text article is mentioned (NUM_ART with TEXT_ANNEX_CC), we retrieve the correct collective agreement via CASE 1 or CASE 2. Then we retrieve the right attached text, using the NUM_LEG and DATE_LEG attached to the TEXT_ANNEX_CC. If there is only one article number corresponding to the NUM_ART, we return this article. Otherwise we apply the disambiguation strategy detailed in the next section.

Most common cases are detailed in Figure 4.

6 ENTITY DISAMBIGUATION

In the event that there are multiple articles with the same number in one collective agreement or attached text, we use an entity disambiguation scheme. The context in which the article is cited will likely be semantically similar to the article as they discuss similar themes. We first embed the whole paragraph where the collective agreement is cited, using a sentence embedding model *SimCSE* [1], trained on French legal documents. *SimCSE* is based on contrastive loss

$$\ell_i = -\log \frac{e^{\text{sim}(\mathbf{h}_i,\mathbf{h}_i^+)/\tau}}{\sum_{j=1}^{N} e^{\text{sim}\left(\mathbf{h}_i,\mathbf{h}_j^+\right)/\tau}},$$

We then also embed each candidate article using the same model, and return the one with the highest cosine similarity with the source paragraph (see cosine similarity distribution on Figure 5). If the score is below threshold (set to 0.3 to eliminate false positives), we output nothing. This allows us to handle 35% of the cases where there are multiple articles with the same number.

7 PERFORMANCE

For the Named Entity Recognition, we achieve a micro (weighted per tag occurence) F1-score of 0,86. The recall (0,90) is slightly higher than the precision, but this is not really an issue, because

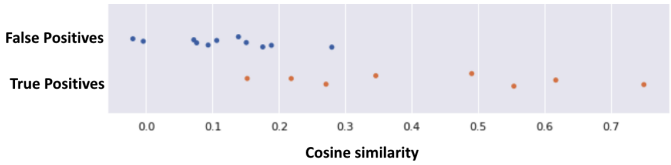


Figure 5: Cosine similarity scores distribution with SimCSE between false and true positives on a small dataset

in most cases, if an entity is wrongly detected, entity matching will not match it to anything. This performance is evaluated on a dataset without the artificially generated paragraphs using the replacement strategy explained in Section 2, in order to evaluate or approach on real data. On the overall extraction pipeline, we achieve a micro (finding all citations, counting duplicates) F1-score of 0.82, and a macro (finding all distinct citations) F1-score of 0,925.

8 CONCLUSION

We proposed an annotation scheme for this task, as well as a dataset boosting strategy to create artificial data. We trained an entity detection model, evaluated, and tested multiple entity linking strategies. Finally, we implemented an entity matching strategy based on different types of citations and achieved great results on a variety of legal documents.

REFERENCES

[1] Tianyu Gao, Xingcheng Yao, and Danqi Chen. 2021. SimCSE: Simple Contrastive Learning of Sentence Embeddings. https://doi.org/10.48550/ARXIV.2104.08821
[2] Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami, and Chris Dyer. 2016. Neural Architectures for Named Entity Recognition. https://doi.org/10.48550/ARXIV.1603.01360
[3] Arie Pratama Sutiono and Gus Hahn-Powell. 2022. Syntax-driven Data Augmentation for Named Entity Recognition. https://doi.org/10.48550/ARXIV.2208.06957

Leveraging Positional Encoding to Improve Fact Identification in Legal Documents

Alexandre G. de Lima
alexandre.lima@ifrn.edu.br
Federal Institute of Rio Grande do Norte
Natal, RN, Brazil

Jose G. Moreno
Mohand Boughanem
Taoufiq Dkaki
firstname.lastname@irit.fr
IRIT, UMR 5505 CNRS, France

Eduardo Henrique da S. Aranha
eduardoaranha@dimap.ufrn.br
Federal University of Rio Grande do Norte
Natal, RN, Brazil

ABSTRACT

Facts are one of the main elements of a legal case and, therefore, their automatic identification is a key step on sub-tasks such as fact-based case search. Facts usually occur at the beginning of case documents. Thus, we hypothesize that the position that each such sentence occupies in its source document can be exploited to improve the performance of fact identification models. To confirm our hypothesis we propose and evaluate models based on sentence content representations and positional encodings. Our results confirm that sentences' positions are valuable information as the best model that exploits content and positional representations outperforms by 7.5%, in terms of F1, the best model that relies only on state-of-the-art representations of sentences.

KEYWORDS

Indian Legal System, Rhetorical role, Deep Learning

1 INTRODUCTION

Legal facts are relevant information for legal professionals such as lawyers and judges. They are valuable to legal assistance tasks such as case search, legal text summarization, legal named entity recognition, and judgment prediction, to mention a few. Systems that are able to automatically extract legal facts and other crucial information from legal texts are of great value to improve and speed up legal processes.

Legal facts (or just facts henceforth) compound the set of legal elements that underlie judicial decisions. To write a legal case, judges usually present facts before presenting reasoning and rulings. So, it is commonplace that facts occur in the first parts of the text of a legal case. Figure 1 shows the distribution of sentences labeled as *Facts* or not in the legal cases of the training dataset exploited in this work. The rightmost histogram aggregates the distribution of 369 documents, while the other ones represent individual documents. Each document is split into ten buckets in order to consider variations in document length. The buckets follow the sequence of sentences in each document, so Bucket 1 relates to the first sentences, while Bucket 10 relates to the last sentences. Histograms in Figure 1 show that, for the considered dataset, the occurrence probability of a *Facts* sentence in the *first parts of a document is higher than in the last ones*. From this insight, we hypothesize that considering sentence position along with state-of-the-art sentence representations may help the identification of sentences with facts. To validate this hypothesis, we perform a series of experiments using state-of-the-art deep learning models.

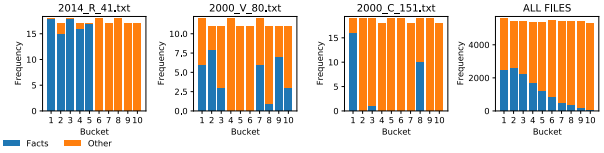


Figure 1: Frequencies of positions (buckets) that sentences occupy in their documents and according to their labels.

Our experiments show that models which rely only on content representations can identify *Facts* sentences to a certain extent, but positional information proves to have a valuable impact in terms of performance: exploiting positional information leads to an improvement of 7.5% when comparing the best model that exploits content and positions, and the best model that exploits content only.

Related Works Several works address the identification of sentences with facts by exploiting machine learning models and implicit/explicit positional information [1–3, 7, 13]. [1–3] employ Recurrent Neural Networks and so their models are able to implicitly exploit positional information. Two works [7, 13] encode sentence position as integer values and exploit them as input data. Our work mainly differs from the cited ones by exploiting sinusoidal encoding methods to represent sentences' positions and pre-trained transformer models to encode sentences' content.

2 POSITION-AWARE CLASSIFICATION

Positional Encoding Positional encoding is a procedure that generates vector representations for each element in a sequence. The Maximum Variances Positional Encoding (mvPE) [9] is a sinusoidal encoding method that aims to produce effective representations by maximizing the variance between consecutive positional encoding vectors through the following equations:

$$mvPE_{(pos,2i)} = \sin(pos \cdot k/m^{2i/e}) \quad (1)$$

$$mvPE_{(pos,2i+1)} = \cos(pos \cdot k/m^{2i/e}) \quad (2)$$

where pos is a position in the sequence, i is a dimension of a $mvPE$ vector such that $0 \leq i < (e-1)/2$, e is the number of dimensions of $mvPE$ vectors, k is a step parameter and m is the max length of the sequence. The k parameter impacts the variance between consecutive $mvPE$ vectors, which gets large as k increases. When $k = 1$, the mvPE method is equal to the positional encoding (PE) of the standard transformer architecture [14].

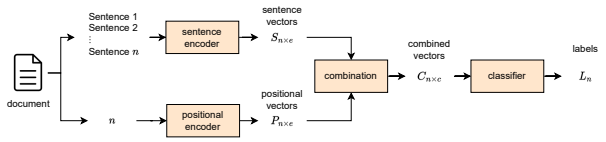


Figure 2: Components and workflow of our position-aware sentence classification framework.

Proposed Framework The proposed position-aware sentence classification framework leverages state-of-the-art sentence representations and the positions that sentences occupy in a document. Figure 2 illustrates the framework whose workflow is the following: the framework is fed with a document split into n sentences; the sentence encoder computes a representation vector for each sentence and it outputs a matrix $S_{n\times e}$, where e is the embedding dimension; from n, the positional encoder computes a positional representation vector for each sentence and it outputs a matrix $P_{n\times e}$; a row-wise combination between $S_{n\times e}$ and $P_{n\times e}$ is made, which results in a matrix $C_{n\times c}$ of combined feature vectors, where c is the resultant dimension of each combined vector; $C_{n\times c}$ is fed to a classifier which outputs n labels. The positional encoder may be PE or mvPE, and the sentence encoder is a transformer-based model. The value of c will depend on the nature of the combination approach ($c = 2e$ for vector concatenation, and $c = e$ for vector sum).

3 EXPERIMENTS AND RESULTS

Dataset We adapted three datasets comprising legal cases from the Indian legal system [2, 8, 10]. We kept the *Facts* labels and replace the other ones with the *Other* label. The final dataset comprises 369 documents and 54,244 sentences in the training set, and 55 documents and 8,046 sentences in the test set. The ratio between *Other* and *Facts* is of about 3.5:1.

Content-based models These models exploit only content representations and each one comprises a sentence encoder and a classifier. We employ BERT base [5], Legal BERT base [4], CaseLaw [15] and SBERT/LaBSE [6, 12] as encoders. The models based on BERT, Legal BERT, and CaseLaw exploit a linear classifier (LC) and are trained by a fine-tuning approach. The models based on SBERT exploit 4 types of classifiers: MLP (Multilayer Perceptron), SVM (Support Vector Machine with a linear kernel), LR (Logistic Regression), and Naïve Bayes. For these models, only the classifiers are updated in the training step.

Combined representation models These are the models that implement the proposed framework. Thus, each model comprises a sentence encoder, a positional encoder, a combination approach, and a classifier. We exploit the same encoders and classifiers of the previous models, and sum and concatenation of vectors as combination approaches.

Implementation details We pick the last hidden state of `[CLS]` token to represent sentences for models based on BERT, Legal BERT, and CaseLaw. In their training, we adopt cross entropy as function loss, the Adam algorithm as optimization method with $2 \cdot 10^{-5}$ as the initial learning rate, and a batch size of 16. The respective linear classifiers are single fully connected layers with a dropout rate of 0.1. We employ 4 fine-tuning epochs. We adopt the SBERT/LaBSE model with default parameters from the Sentence Transformers library[1] to implement SBERT sentence encoders. Regarding positional encoding methods, we set $m = 10,000$ for PE and mvPE and $k = 1500$ for mvPE. For the other classifiers, we adopt the implementations from Scikit-learn library[2] [11]. In general, we adopt the default hyperparameter values with the following exceptions: `early_stopping=True` for MLP; and `solver="sag"` and `max_iter=200` for Logistic Regression.

Evaluation Results are reported through Precision, Recall, and F1 scores and by taking *Facts* as the positive label. All reported values correspond to the average performances of models over the test set and five executions with different seeds. For fine-tuned models, we report the scores from the best fine-tuning epoch in terms of F1.

Discussion We analyze the effects of positional information exploitation by comparing the scores in each line of Tables 1, 2, and

Table 1: Precision scores achieved by exploited models. The best score is formatted in bold. (S) and (C) stand respectively for sum and concatenation.

Sentence encoder + classifier	Content-based model	Combined representation model			
		PE(S)	PE(C)	mvPE(S)	mvPE(C)
BERT + LC	0.647	0.661	0.687	0.571	0.626
Legal BERT + LC	0.668	0.669	**0.692**	0.577	0.597
CaseLaw + LC	0.636	0.649	0.609	0.612	0.575
SBERT + MLP	0.611	0.649	0.669	0.636	0.629
SBERT + SVM	0.663	0.671	0.599	0.530	0.635
SBERT + LR	0.661	0.672	0.672	0.646	0.668
SBERT + Naïve Bayes	0.417	0.343	0.354	0.405	0.448

Table 2: Recall scores achieved by exploited models. The best score is formatted in bold. (S) and (C) stand respectively for sum and concatenation.

Sentence encoder + classifier	Content-based model	Combined representation model			
		PE(S)	PE(C)	mvPE(S)	mvPE(C)
BERT + LC	0.548	0.605	0.586	0.560	0.510
Legal BERT + LC	0.496	0.596	0.563	0.552	0.521
CaseLaw + LC	0.577	0.612	0.699	0.523	0.643
SBERT + MLP	0.515	0.483	0.508	0.410	0.520
SBERT + SVM	0.411	0.424	0.539	0.315	0.475
SBERT + LR	0.412	0.473	0.476	0.320	0.474
SBERT + Naïve Bayes	0.710	0.759	**0.771**	0.582	0.723

Table 3: F1 scores achieved by exploited models. The best score is formatted in bold. (S) and (C) stand respectively for sum and concatenation.

Sentence encoder + classifier	Content-based model	Combined representation model			
		PE(S)	PE(C)	mvPE(S)	mvPE(C)
BERT + LC	0.591	0.626	0.621	0.563	0.545
Legal BERT + LC	0.556	0.626	0.604	0.560	0.549
CaseLaw + LC	0.603	0.623	**0.649**	0.562	0.603
SBERT + MLP	0.557	0.553	0.575	0.497	0.568
SBERT + SVM	0.506	0.517	0.554	0.393	0.543
SBERT + LR	0.508	0.554	0.556	0.427	0.553
SBERT + Naïve Bayes	0.524	0.473	0.485	0.478	0.553

[1] 2.2.0 version
[2] 0.24.1 version

3 and by considering each content-based model as the reference model of the respective line. We see that PE(S) improves Precision and Recall of nearly all reference models, whose SBERT+NB and SBERT+MLP are the respective exceptions. As consequence, most reference models improve F1. PE(C) improves Precision of four reference models (BERT+LC, Legal BERT+LC, SBERT+MLP, SBERT+LR) and improves Recall of all reference models except for SBERT+MLP. Even though, PE(C) improves F1 of almost all reference models (SBERT+NB is the exception). That means that the Recall gains are enough to improve F1 in most cases. We see that mvPE(S) performs poorly. It improves Precision of one reference model (SBERT+MLP), Recall of two reference models (BERT+LC and Legal BERT+LC), and F1 of one model (Legal BERT+LC). mvPE(C) improves Precision of three reference models (SBERT+MLP, SBERT+LR, and SBERT+NB), Recall of almost all reference models (BERT+LC is the exception), and F1 of four reference models (SBERT+MLP, SBERT+SVM, SBERT+LR and SBERT+NB). The results show a better performance of PE when compared to mvPE. This is intriguing since mvPE was devised to produce better representations than PE. Because we do not look for an optimal value of k, the chosen value may be the source of the lower performance of mvPE-based models. Remarkably, PE always improves fine-tuned models (BERT, Legal BERT, and CaseLaw): the smallest gain is 3.5% for CaseLaw+LC+PE(S) and the largest one is 12.5% for Legal BERT+LC+PE(S). Regarding combination approaches, concatenation is clearly superior when we regard mvPE. For PE, both approaches are good and the concatenation one appears to be a little better.

4 CONCLUSIONS

This paper presents a new strategy to identify facts from legal cases. We noticed that sentences with facts often occur at the beginning of a document and therefore we hypothesized that we could leverage the position that each sentence occupies in its source document to improve the performance of models. Our results show that the exploitation of sentences' positions encoded by PE is an efficient strategy to improve the performance of sentence classification models (the best average F1 score increased from 0.603 to 0.649). We deem that the improvement occurs because there is a correlation between sentences' positions and sentences labeled as *Facts*. Hence, the proposed strategy is limited to datasets that present some correlation degree regarding sentences' positions and labels. When this is not the case, we believe that the proposed strategy will lead to minimal or any improvements. It may even harm models' performance since the positional information can work like a noise signal.

ACKNOWLEDGEMENTS

This work has been supported by the Coordination for the Improvement of Higher Education Personnel - Brazil (CAPES) and was partially supported by the LawBot project (ANR-20-CE38-0013), granted by ANR the French Agence Nationale de la Recherche.

REFERENCES

[1] S R Ahmad, D Harris, and I Sahibzada. 2020. Understanding Legal Documents: Classification of Rhetorical Role of Sentences Using Deep Learning and Natural Language Processing. In *ICSC*.

[2] P Bhattacharya, S Paul, K Ghosh, S Ghosh, and A Wyner. 2019. Identification of Rhetorical Roles of Sentences in Indian Legal Judgments. In *JURIX*.

[3] P Bhattacharya, S Paul, K Ghosh, S Ghosh, and A Wyner. 2021. DeepRhole: deep learning for rhetorical role labeling of sentences in legal case documents. *Artificial Intelligence and Law* (2021).

[4] I Chalkidis, M Fergadiotis, P Malakasiotis, N Aletras, and I Androutsopoulos. 2020. LEGAL-BERT: "Preparing the Muppets for Court'". In *Findings of EMNLP*.

[5] J Devlin, M-W Chang, K Lee, and K Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *NAACL*.

[6] Fangxiaoyu Feng, Yinfei Yang, Daniel Cer, Naveen Arivazhagan, and Wei Wang. 2022. Language-agnostic BERT Sentence Embedding. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Association for Computational Linguistics, Dublin, Ireland, 878–891. https://doi.org/10.18653/v1/2022.acl-long.62

[7] B Hachey and C Grover. 2004. A Rhetorical Status Classifier for Legal Text Summarisation. In *Text Summarization Branches Out*.

[8] P Kalamkar, A Tiwari, A Agarwal, S Karn, S Gupta, V Raghavan, and A Modi. 2022. Corpus for Automatic Structuring of Legal Documents. In *LREC*.

[9] H Li, Y.C. Wang, A, Y Liu, D Tang, Z Lei, and W Li. 2019. An Augmented Transformer Architecture for Natural Language Generation Tasks. In *ICDMW*.

[10] V Malik, R Sanjay, S K Nigam, K Ghosh, S K Guha, A Bhattacharya, and A Modi. 2021. ILDC for CJPE: Indian Legal Documents Corpus for Court Judgment Prediction and Explanation. In *ACL/IJCNLP*.

[11] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research* 12 (2011), 2825–2830.

[12] N Reimers and I Gurevych. 2019. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. In *EMNLP 2019*.

[13] O Shulayeva, A Siddharthan, and A Z Wyner. 2017. Recognizing cited facts and principles in legal judgements. *Artif. Intell. Law* (2017).

[14] A Vaswani, N Shazeer, N Parmar, J Uszkoreit, L Jones, A N. Gomez, L Kaiser, and I Polosukhin. 2017. Attention is All you Need. In *NeurIPS*.

[15] L Zheng, N Guha, B R. Anderson, P Henderson, and D E. Ho. 2021. When Does Pretraining Help? Assessing Self-Supervised Learning for Law and the CaseHOLD Dataset. In *ICAIL 2021*.

Semantic Search in Legislation

Adeline Nazarenko*
François Lévy
Haïfa Zargayouna
LIPN – Université Sorbonne Paris Nord & CNRS
Villetaneuse, France
firstname.lastname@lipn.univ-paris13.fr

Adam Wyner
Department of Computer Science, Swansea University
Swansea, United Kingdom
a.z.wyner@swansea.ac.uk

KEYWORDS

Semantic annotation, semantic information retrieval, legal text mining, GDPR

ACM Reference Format:
Adeline Nazarenko, François Lévy, Haïfa Zargayouna, and Adam Wyner. 2023. Semantic Search in Legislation. In *Proceedings of Legal Information Retrieval (LegalIR)*. ACM, New York, NY, USA, 3 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

One of the main early objectives of AI and Law [5] has been to analyse legislation and regulations to allow for querying and reasoning. Progress has been made to improve interoperability at the document and rule levels, *e.g.* with OASIS standards AkomaNtoso[1] and LegalRuleML[2], but much remains to be done to develop a real legal semantic web that can be queried for substantive content. Works have been done to associate formal rules with texts [6], but it is difficult to define a sound methodology to formalize legislative provisions on a large scale and meet generic needs.

We consider that legal information retrieval [3] represents a promising alternative approach, which is more practical on a large scale and more flexible to address the diversity of legal needs (*e.g.* retrieving, clustering, comparing and contextualizing provisions). Actually, legal public services[3] do not offer advanced semantic search functionalities that could facilitate the daily work of legal professionals as well as contribute to the progress of the formal approaches in the long term.

This paper proposes a semantics approach, which relies on a lightweight, coarse-grained, interpretation-neutral, semantic description of legal provisions (Sec. 2) and a search strategy combining keywords and semantics (Sec. 3). It also reports on a proof-of-concept experiment on the GDPR[4] (Sec. 4) with examples of queries that can be better answered by semantic than by plain text search. This experimentation demonstrates the potential of the proposed approach and provides a basis for further development in legal information retrieval.[5]

2 ENRICHING LEGAL PROVISIONS WITH SEMANTIC METADATA

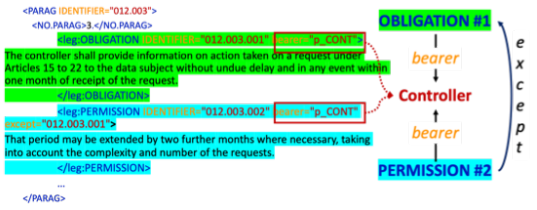
The proposed approach relies on the CLAL[6] language which is designed to enrich the legal source with a coarse-grained, interpretation-neutral, semantic annotation layer [4]. The language is used to annotate the text itself, making explicit the role of the pieces of text in the construction of legal meaning.

Four kinds of information are annotated:

- Main concepts and actors: `concept`, `person`, `legal_body`;
- Categories of statements (see Table 1);
- Relations between statements and concepts or actors: `bearer`, `target` and `obj` roles;
- Relations between statements: `rel`, `except` and `reparation`.

The annotation, which is encoded in XML, can be represented as a semantic graph over textual provisions (Fig. 1). The vocabulary

Figure 1: Portion of the semantic graph built above the GDPR. The identifiers refer to sentence numbers.



of CLAL has been designed for the GDPR experiment. Even if it were to be adapted or extended to account for additional texts, the language should be kept small and manageable.

The quality, consistency, and stability of the annotations can be ensured provided a good annotation methodology [1] is followed.

3 RETRIEVING PROVISIONS BASED ON SEMANTICS

The annotations are mainly descriptive of textual passages. To retrieve passages based on semantics, we query the XML representation of annotations with a query language. In order to improve

*Nazarenko, Lévy, and Wyner contributed equally to this research. Zargayouna contributed expertise in information retrieval.
[1]http://www.akomantoso.org/
[2]https://www.oasis-open.org/committees/tc_home.php?wg_abbrev=legalruleml
[3]Such as https://www.legifrance.gouv.fr or www.legislation.gov.uk.
[4]The European General Data Protection Regulation https://gdpr-info.eu/.

LegalIR, April 02, 2023, Dublin, Ireland

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM…$15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

[5]The CLAL language as well as all resources developed for that GDPR experiment publicly available (https://lipn.univ-paris13.fr/~fl/CLAL/).
[6]The Core Legal Annotation Language is formalized in XML and described in the XSD schema language.

Table 1: CLAL statement types and subtypes.

Prescriptive stat.	*Constitutive stat.*	*Dependent stat.*
obligation	definition	exception
prohibition	attribution*	complement*
permission	competency	procedure
power*	responsibility	text-specification
executive	quality	precision
ruling		impact
right		validity
		reparation

precision and recall, we focus on two approaches to query expansion using semantic annotations: semantic axioms and exploration with semantic dependency relations. Semantic search complements query expansion using REGEX or additional query words, whicih can be integrated with semantic search as reported below and in [7].

The overall querying process is the following:

- The user query is translated in a SQL-like form, using either a LN2SQL translator or a dedicated interface, using a combination of keywords and semantic concepts.
- The query can be expanded into a set of queries based on semantic axioms or semantic dependency relations.
- The statements that match both the keywords and semantic concepts of the queries are returned to the user.

3.1 Query Exploration

As long discussed in the literature, there are axiomatic semantic relationships between legal concepts [2] such as the following, expressed as implications:

(1) Y bears an obligation with respect to X ⇒ X has a right with respect to Y.
(2) It is prohibited to do X except if Y ⇒ It is permitted to do X only if Y

In relation to information retrieval, these imply that when one searches for an expression with the annotation of `obligation`, then one should also always retrieve those expressions with the annotation of `right`. Moreover, for example, the bearer of the `obligation` should be the target of the `right`

3.2 Exploration with Semantic Relations

An additional approach to query expansion is to utilise semantic *dependency relations* between statements, which return graphs of statements in the given relation, *e.g.* `complement:procedure`, `exception`, `reparation`, etc., rather than isolated statements. In other words, searching for obligation statements can also optionally (at the discretion of the user) return those statements which are in specific relations to those obligations, e.g., express a procedure, an exception, or a reparation.

For example, in looking for an obligation, we would also like to find those expressions which more precisely characterise the obligation; that is, along with (1) we should like to find statements such as (2) that are in a relation to (1).

(1) The controller shall document any personal data breaches, comprising the facts relating to the personal data breach, its effects and the remedial action taken. `OBLIGATION`
(2) That documentation shall enable the supervisory authority to verify compliance with this Article.
`COMPLEMENT: PRECISION`

By such queries, one can generate or query a graph of a statement (or statements) with other statements with which they are in semantic relations.

4 AN EXPERIMENT ON GDPR

The GDPR annotation was achieved with a short SPIN project.[7] 6 law students enriched the GDPR with semantic tags following semantic and technical guidance. The adjudication then delivered a reliable and consensual annotation of the GDPR. This experiment globally validates our annotation approach and shows that annotation can be quickly achieved at a reasonable cost.[8]

To show the advantages of semantic annotation for search, we designed a small experimental search engine based on an SQL-like querying language combining semantic and plain-text criteria, and we illustrate it on few test questions focusing on *obligations*

Q1. What are the rights of the data subject? The translation of the question in a semantic query is straightforward: "Select the statements annotated as `right` which `bearer` role is filled by the identifier of the data subject". It returns 19 statements[9] whereas searching for sentences containing the strings “right” and “data subject” provides 50 additional noisy ones[10]. In this case, semantic search is more precise than full text search, an advantage for legal practitioners who have to browse large quantities of legal sources.

Q2. What are the obligations of a data controller? The question seems to translate directly into a semantic query: “Select the statements annotated as `obligation` with the `bearer` role filled by the identifier of the data controller”. However, as stated in axiom (1), obligations imply rights , which gives rise to a second query (*query exploration*): “Select the statements annotated as `right` with a `target` role filled by the identifier of the data controller”. This double query returns 64 obligations and 17 rights. In comparison, a plain text search (*Which sentences contain both “controller” and “obligation” keywords?*) gives 19 statements, among which only one right and two obligations are relevant. This is due to the fact that obligations are not expressed with the word "obligation" (nor any similar keyword). It is also important to filter out the statements in which the "data controller" is mentioned but not in the proper role.

Q3. What are the obligations of the controller in case of data breach? *Q*3 translates into a hybrid query combining semantic criteria and keywords: “Select statements annotated as `obligation` with the `bearer` role filled by the identifier of the data controller and containing the string ‘data breach’ ”. This query returns 3 statements. Interestingly, thanks to the capabilities to explore the semantic annotations, Art. 34 §1[11] is returned along with two exceptions

[7] Swansea Paid Internship Network funding provided by Swansea University.
[8] The 99 articles of the GDPR were reliably annotated in less than 20 hours without legal professionals.
[9] Such as *the right to obtain from the controller, without undue delay the rectification of inaccurate personal data concerning him or her.*
[10] Such as Art. 12 § 2 "The controller shall facilitate the exercise of data subject rights under Articles 15 to 22"
[11] *When the personal data breach is likely to result in a high risk to the rights and freedoms of natural persons, the controller shall communicate the personal data breach to the data subject without undue delay.*

that are relevant for answering $Q3$. Those exceptions are retrieved using the query expansion mechanism and one would probably escape the reader otherwise because it is textually distant (Art. 23).

5 CONCLUSION

This experience on GDPR shows that it is possible to enrich legislation with a generic and shallow semantic layer that nevertheless supports advanced and valuable search functionality for anyone looking to explore legal texts.

There is still work to be done to operationalise, evaluate, augment querying, and make it user-friendy. Given that the main issues in creating the corpus, namely scalability and quality of annotation, have been largely addressed, research can focus on information retrieval. The SPIN project has shown that this semantic approach is realistic and promising as a basis for legal text retrieval and mining.

REFERENCES

[1] Karën Fort. 2016. *Collaborative Annotation for Reliable Natural Language Processing: Technical and Sociological Aspects.* Wiley-ISTE. 196 pages.

[2] Wesley Hohfeld. 1923. Fundamental legal conceptions applied in judicial reasoning. In *Fundamental Legal Conceptions Applied in Judicial Reasoning and Other Legal Essays*, W. Cook (Ed.). Yale University Press, 23–64.

[3] K. Tamsin Maxwell and Burkhard Schafer. 2008. Concept and Context in Legal Information Retrieval. In *Oric. of the 21st JURIX*. IOS Press, 63–72.

[4] Adeline Nazarenko, François Lévy, and Adam Wyner. 2021. A Pragmatic Approach to Semantic Annotation for Search of Legal Texts - An Experiment on GDPR. In *Legal Knowledge and Information Systems - JURIX*, Schweighofer Erich (Ed.). IOS Press, 23–32. https://doi.org/10.3233/FAIA210313

[5] Ronald Stamper. 1980. LEGOL: Modelling Legal Rules by Computer. *Computer Science and Law* (1980), 45–71.

[6] Adam Z. Wyner. 2015. From the Language of Legislation to Executable Logic Programs. In *Logic in the Theory and Practice of Lawmaking*, Michal Araszkiewicz and Krzysztof Pleszka (Eds.). Legisprudence Library, Vol. 2. Cham: Imprint: Springer, 409–434. https://doi.org/10.1007/978-3-319-19575-9_15

[7] Adam Z. Wyner, Fraser Gough, François Lévy, Matt Lynch, and Adeline Nazarenko. 2017. On Annotation of the Textual Contents of Scottish Legal Instruments. In *Legal Knowledge and Information Systems - JURIX 2017: The Thirtieth Annual Conference, Luxembourg, 13-15 December 2017 (Frontiers in Artificial Intelligence and Applications, Vol. 302)*, Adam Z. Wyner and Giovanni Casini (Eds.). IOS Press, 101–106.

An Annotation Framework for Benchmark Creation in the Legal Case Retrieval Domain

Tobias Fink
tobias.fink@tuwien.ac.at
Technische Universität Wien
Research Studios Austria FG
Vienna, Austria

Yasin Ghafourian
yasin.ghafourian@researchstudio.at
Research Studios Austria FG
Technische Universität Wien
Vienna, Austria

Georgios Peikos
georgios.peikos@unimib.it
University of Milano-Bicocca
Milan, Italy

Florina Piroi
florina.piroi@researchstudio.at
Research Studios Austria FG
Technische Universität Wien
Vienna, Austria

Allan Hanbury
allan.hanbury@tuwien.ac.at
Technische Universität Wien
Vienna, Austria

ACM Reference Format:
Tobias Fink, Yasin Ghafourian, Georgios Peikos, Florina Piroi, and Allan Hanbury. 2018. An Annotation Framework for Benchmark Creation in the Legal Case Retrieval Domain. In *Proceedings of The first international workshop on Legal Information Retrieval (ECIR '23).* ACM, New York, NY, USA, 3 pages.

1 INTRODUCTION

The increasing digitalization of legal data leads to a greater demand for legal information retrieval (LIR) systems. Especially retrieval of precedent or notice court cases is an active research topic. However, non-English benchmarks in this field are scarce, and although legal cases are very long, when a precedent or notice case is cited, the citation is on document level instead of passage level. As a result, it needs to be clarified which passages of the cited case lead to its citation. Similarly, in existing benchmarks for legal case retrieval, for queries in the form of legal questions, summary or whole case relevance, annotations are typically only available for the document level [2, 4, 6]. Some collections feature relevant passage annotations on a smaller scale, e.g. COLIEE case entailment [6].

This paper describes a framework for building a benchmark collection for passage retrieval in the legal domain. It involves three steps: data collection, case entailment annotation, and query relevance annotation for legal case retrieval. Using the proposed framework, we build a court case retrieval benchmark in the domain of building regulations. The purpose of this paper is to comment on the framework's first and second steps and provide information about the implementation and planned use of its final step.

2 FRAMEWORK DESCRIPTION

Step 1: The first step of the framework involves collecting court cases that need to fulfil the following requirements to be suitable: Firstly, the court case texts need to be digitally and publicly available. Also, it is required that these cases contain passages that cite other court cases, i.e., citing passages. As is commonly the case, these passages only contain the ID of the cited case. Lastly, the citations need to be valid, i.e. it needs to be possible to automatically extract the cited case texts from the collection as well. The citations can only be used if they match valid document IDs within the collection. To create a German language benchmark collection, respecting the requirements mentioned above, we exploit data from the Austrian RIS (Rechtsinformationssystem)[1]. In detail, we collect court cases of the Austrian Verwaltungsgerichtshof (Vwgh - Supreme Administrative Court) in XML format using the API and restrict them to the building regulations domain using RIS metadata. The specific information about the created collection is presented in Table 1. By extracting passages from the XML formatted files, cases are converted to plain text. We employ regular expression patterns to detect the passages that cite other cases and automatically identify the matching cited case documents in the collection.

Table 1: Statistics for our dataset of Austrian court cases in the Viennese building regulations domain.

Statistic	Value
Cases	1703
Passages	87,149
Mean Passages per Case	51.17
Citing Passages	6893
Cited Cases	8940
Mean Cited Cases per Case Citation	1.3
Annotated Passages	4,146
Relevant Passages	547

Step 2: In the second step of the framework, passages from the original case that cite other cases are randomly selected. Hereafter, we refer to such an extracted passage as topic, T. A topic T, along with its cited case, is shown to expert annotators from the building regulations domain in a simple UI[2]. For a given topic T, the annotators are asked to determine which passages $c_1, ..., c_k$ of the cited case C lead to its citation, i.e. entails the topic $entails(c_i, T)$. These entailing passages are referred to as relevant passages $\{R \subseteq C | r_i \in R \wedge entails(r_i, T)\}$. This process leads to a benchmark collection for case entailment. We have randomly selected 64 cases from the collection and annotated passages for a total of 64 topics T. Since some of the topics cite multiple court cases, relevant passages have been

ECIR '23, April 02–06, 2023, Dublin, Ireland
2018. ACM ISBN 978-1-4503-XXXX-X/18/06…$15.00

[1] https://www.ris.bka.gv.at/ and https://data.bka.gv.at/ris/api/v2.6/
[2] https://labelstud.io/

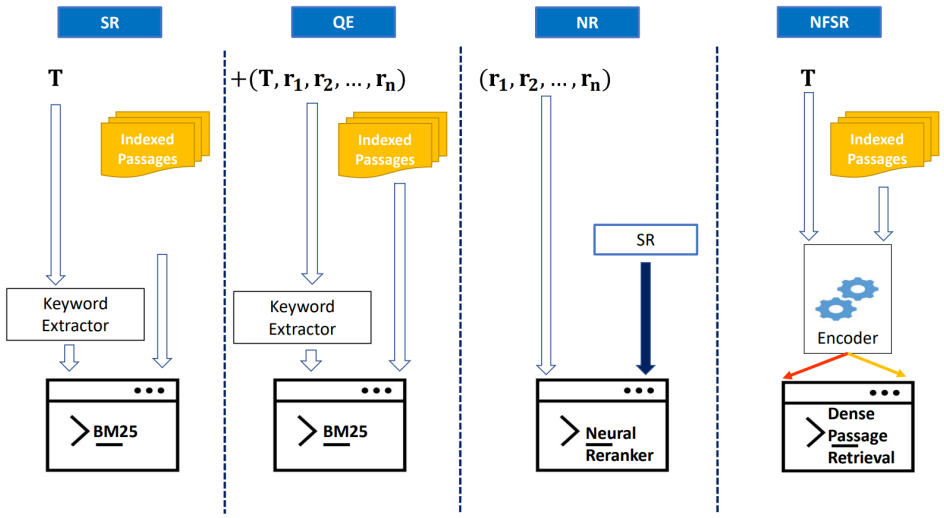


Figure 1: An overview of the third step of the framework where the candidate passages for pooling and annotation are selected.

annotated in 85 cited cases. Already with this step a benchmark collection for case entailment was created.

Step 3: In the framework's third step, we aim to expand the scope of the created benchmark collection by manually annotating more relevant passages for a given topic T. For this purpose, all passages in the collection are indexed by an IR system and queried using the topics T to create a new pool of passages to be annotated. We employ different IR systems to query the collection, and each system will add its top-ranked k candidate passages to the annotation pool for a topic T. Duplicate passages will be removed. We suggest to use at least four main retrieval system architectures (see Figure 1) for retrieval that leverage different topic representations and retrieval models, to ensure relevance and diversity. Some of these architectures should utilize the relevant passages $r_1, ..., r_n \in R$ during retrieval, as described below.

Standard Retrieval (SR). This architecture uses a lexical model (such as BM25) as its retrieval algorithm. A keyword extraction method extracts the highest-scoring keywords from T to formulate a query, an ad-hoc representation of passage. To that aim, the Kullback-Leibler divergence for Informativeness (KLI) can be used as it is an effective method for extracting keywords in the legal domain [1, 7].

Query Expansion (QE). This system is an alteration of SR. This architecture leverages the already annotated relevant passages $r_1, ..., r_n$ and the original topic T. Specifically, the texts of T and $r_1, ..., r_n$ are concatenated, and the keywords extracted as in SR.

Neural Reranking (NR). This is a two-step retrieval architecture, composed using SR as first-stage retrieval and passage reranking with BERT [5]. First, the passages $sr_1, ..., sr_m$ are retrieved following the SR approach, producing a ranking of passages for a topic T. Then, for each of the relevant passages $r_1, ..., r_n$, BERT is applied to score and rank $\{sr_1, ..., sr_m | BERT(r_i, sr_j)\}$. These n rankings are then aggregated, by summing the scores or reciprocal ranks per passage, to create a final ranking of passages.

Neural First Stage Retrieval (NFSR). This architecture will use Dense Passage Retrieval (DPR) [3] to encode both topic T and candidate passages from the collection as vectors for which a cosine similarity will be calculated to produce a ranking of passages. The model is trained on the Step 2 benchmark dataset.

Multiple instances of the systems can be used depending on the number of employed annotators. These systems can leverage a variety of staticical retrieval models, keyword extraction methods, and re-ranking models.

As for our benchmark dataset is concerned, we plan to implement each of the described architectures for pooling. However, since at this point it is not clear how many annotators will be available for Step 3, we have not fixed the number of system instances yet.

3 CONCLUSIONS

We introduced a 3-step framework that combines multiple data extraction and annotation methods to create a benchmark collection for the legal domain. The created benchmark collection will be suitable both for legal case retrieval and for legal passage retrieval. In addition, the collection can be used to compare the effectiveness of query extraction techniques, as all topics are text passages. As the relevant passages are annotated based on their legal relevance, i.e. the annotators were asked to determine which parts of the cited case lead to its citation in the original case, the collection also acts as a benchmark for case entailment.

REFERENCES

[1] Askari, A., Peikos, G., Pasi, G., and Verberne, S. LeiBi@ COLIEE 2022: Aggregating tuned lexical models with a cluster-driven BERT-based model for case law retrieval. *arXiv preprint arXiv:2205.13351* (2022).

[2] Bhattacharya, P., Ghosh, K., Ghosh, S., Pal, A., Mehta, P., Bhattacharya, A., and Majumder, P. Overview of the FIRE 2019 AILA track: Artificial intelligence for legal assistance. P. Mehta, P. Rosso, P. Majumder, and M. Mitra, Eds., vol. 2517 of *CEUR Workshop Proceedings*, CEUR-WS.org, pp. 1–12.

[3] Karpukhin, V., Oğuz, B., Min, S., Lewis, P., Wu, L., Edunov, S., Chen, D., and Yih, W.-t. Dense passage retrieval for open-domain question answering. *arXiv preprint arXiv:2004.04906* (2020).

[4] Locke, D., and Zuccon, G. A test collection for evaluating legal case law search. In *SIGIR 2018, Ann Arbor, MI, USA, July 08-12, 2018* (2018), K. Collins-Thompson, Q. Mei, B. D. Davison, Y. Liu, and E. Yilmaz, Eds., ACM, pp. 1261–1264.

[5] Nogueira, R., and Cho, K. Passage re-ranking with BERT. *arXiv preprint arXiv:1901.04085* (2019).

[6] Rabelo, J., Kim, M., Goebel, R., Yoshioka, M., Kano, Y., and Satoh, K. A summary of the COLIEE 2019 competition. In *New Frontiers in Artificial Intelligence - JSAI-isAI International Workshops, JURISIN, AI-Biz, LENLS, Kansei-AI, Yokohama, Japan, November 10-12, 2019, Revised Selected Papers* (2019), M. Sakamoto, N. Okazaki, K. Mineshima, and K. Satoh, Eds., vol. 12331 of *Lecture Notes in Computer Science*, Springer, pp. 34–49.

[7] Verberne, S., Sappelli, M., Hiemstra, D., and Kraaij, W. Evaluation and analysis of term scoring methods for term extraction. *Information Retrieval Journal 19*, 5 (2016), 510–545.

Exploring Semi-supervised Hierarchical Stacked Encoder for Legal Judgement Prediction

Nishchal Prasad
IRIT, Toulouse, France
Nishchal.Prasad@irit.fr

Mohand Boughanem
IRIT, Toulouse, France
Mohand.Boughanem@irit.fr

Taoufiq Dkaki
IRIT, Toulouse, France
Taoufiq.Dkaki@irit.fr

ABSTRACT

Predicting the judgment of a legal case from its unannotated case facts is a challenging task. The lengthy and non-uniform document structure poses an even greater challenge in extracting information for decision prediction. In this work, we explore and propose a two-level classification mechanism; both supervised and unsupervised; by using domain-specific pre-trained BERT to extract information from long documents in terms of sentence embeddings further processing with transformer encoder layer and use unsupervised clustering to extract hidden labels from these embeddings to better predict a judgment of a legal case. We conduct several experiments with this mechanism and see higher performance gains than the previously proposed methods on the ILDC dataset. Our experimental results also show the importance of domain-specific pre-training of Transformer Encoders in legal information processing.

KEYWORDS

Domain Specific Pre-trained Transformers, Two-level Classification Mechanism, Semi-supervised Learning

1 INTRODUCTION

Automating legal case proceedings can assist the decision-making process with speed and robustness, which can save time and be beneficial to both the legal authorities and the people concerned. One of the underlying tasks which deal with this broader aspect is the prediction of the outcome for the legal cases with just the facts of the case, which depicts the general real-life setting where only the case facts are provided. For this problem, many techniques have been explored in the past using machine learning to predict the outcome of legal cases.

For their Case Judgment Prediction and Explanation (CJPE) task, Malik et al. [2] introduced the Indian Legal Document Corpus (ILDC) dataset which reflects our ideal general setting for legal case documents. We use this dataset for testing our methods and compare them with other state-of-the-art models on the same. In our past work [6], we demonstrated that a domain-specific pre-trained model can perform noticeably better and adapt effectively to domains of the same kind with different syntax, lexicon, and grammatical settings. Shounak et al.[5] pre-trained BERT on a large corpus of Indian legal documents and applied it to several benchmark legal NLP tasks over both Indian legal text and those belonging to other countries. One problem with a BERT-based transformer architecture is the constraint in processing large documents due to the input limit of 512 tokens. In this work, we aim to predict decisions from large and non-uniform structured legal documents having very low annotations (i.e. just the prediction label). We explore the effects of some of the available legal (i.e domain-specific) pre-trained BERT models with an unsupervised clustering algorithm (HDBSCAN[3]) and propose a method, that leverages both of these techniques to understand long and unstructured legal case documents.

2 METHOD

We modify the method of Hierarchical Transformer[4] to tackle this problem of large document processing with the use of clustering to be able to extract more information for further processing. We experiment with two domain-specific pre-trained BERT models (LEGAL-BERT[1] and InLegalBERT[5]) with the hypothesis that domain pre-training of a transformer model is necessary for the in-domain vocabulary and lexical understanding [6]. We process the documents in two steps (figure 1). We divide the document into parts called chunks (sequential sets of words). We tokenize and wrap these chunks with the [CLS] and [SEP] tokens. These tokenized chunks with their respective document label individually form input to a BERT model for fine-tuning (step I, fig. 1).After fine-tuning, the [CLS] token embeddings are extracted for individual chunks which are used for the next step of processing. The [CLS] embeddings are considered here to be a representation of the chunk, and concatenating them together gives an approximate representation of the whole document.

In step II, We use transformer layers on the extracted [CLS] embeddings for the intra-chunk attention to learn the whole document representation. We also experiment with RNNs (BiLSTM, GRU) after the transformer encoder (Table 1). The [CLS] embeddings are also used for the unsupervised learning mechanism i.e. clustering the individual chunks which are used as extra information while training. This provides information for the unlabeled parts of the document, i.e. which partly relates to which topic. These individual cluster features along with the chunk embeddings help the model to better understand its contents and also add the constituent information of the related and unrelated parts of the document. For example, two chunks relating to the same law article in two different documents will be grouped together while clustering, and this grouping will be used as a piece of extra information extracted from the document. We have experimented with two variants of the inputs to the Transformer Encoder layer: The [CLS] chunk embeddings extracted from the finetuned BERT (α), or the dimension-reduced [CLS] chunk embeddings (β) from pUMAP[1] having 64 dimensions. Table 1 shows the impact of these two combinations on the classification performance. For clustering, we use HDBSCAN[3] with a minimum cluster and sample size of 15 and 10 respectively.

3 DATASET AND RESULTS

To conduct the experiments, we used the Indian Legal Document Corpus ILDC[2] to replicate a real-life setting of decision prediction of legal documents, for our proposed method. The dataset consists

[1] https://umap-learn.readthedocs.io/en/latest/parametric_umap.html

Table 1: Experimental results of legal text classification on ILDC dataset for different architectures

Models (e = epochs)		Metrics (%)					
		Validation			Test		
		Acc.	mP	mR	Acc.	mP	mR
Pre-Trained Transformer Encoders (fine-tuned)							
BERT[6] e=2		-	-	-	60.52	66.13	60.55
XLNet[6] e=2		-	-	-	70.51	72.01	70.09
LEGAL-BERT[6] e=2		-	-	-	73.83	73.90	73.84
InLEGAL-BERT e=4		76.15	76.87	76.16	76.00	76.17	76.02
InLEGAL-BERT + BiGRU [5]		-	-	-	-	83.43	83.15
Two-level Architectures:							
LEGAL-BERT+ ($fine\text{-}tuned$) $e = 4$	Bi-LSTM + Dropout e=6 [6]	-	-	-	80.60	0.8106	80.63
	Bi-LSTM + Dropout +Multi-head attention$_\beta$ e=6 [6]	-	-	-	80.90	81.60	80.90
InLEGAL-BERT+ ($fine\text{-}tuned$) $e = 4$	2×Bi-GRU e=3	83.37	83.35	83.28	83.31	83.39	83.30
	Bi-LSTM + Bi-GRU e=3	83.97	83.40	83.25	83.11	83.76	83.09
	1× Encoder e=3	84.10	84.33	84.10	**83.72**	**83.74**	**83.72**
InLEGAL-BERT+ ($fine\text{-}tuned$) $e = 4$ +pUMAP+HDBSCAN	(α) 1×Encoder e=1	**84.51**	**84.56**	**84.51**	**83.65**	**83.66**	**83.65**
	(β) 1× Encoder e=1	83.90	84.01	83.90	83.39	83.39	83.39
	(α) 1× Encoder + BiLSTM e=3	**85.01**	**85.03**	**85.01**	83.59	83.59	83.58

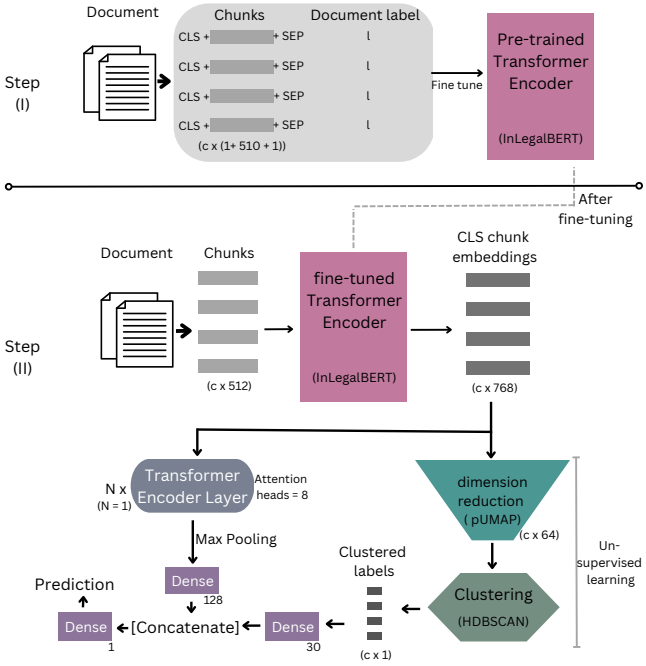


Figure 1: Two-level classification architecture

of 39898 case proceedings (in English) from the Supreme Court of India (SCI). Each document is identified with the initial judgment rendered by the SCI judge(s) between 'rejected' and 'accepted'. Hence, our task of decision prediction can be formulated as a binary text classification problem. The dataset is pre-divided into a test (1517 documents) and validation (994 documents) set, we use the same for our experiments.

We show concise results in Table 1 amongst the experiments conducted with different architectures. We used accuracy, macro-precision (mP), and macro-recall (mR), as the performance metrics and compare them with the previous baseline models. The InLegalBERT with RNNs performs 3 points higher than LEGAL-BERT, showing the effectiveness of further in-domain pre-training. The RNNs give almost the same performance as the Transformer Encoder layers in the test set, but the Encoders were more stable while training by showing marginal variations (≈ 0.1) in the validation metrics for a set of 3-4 subsequent epochs. Thus we chose the encoder layers to further learn from the [CLS] chunk embeddings. The effect of the unsupervised clustering mechanism with its combinations with the Transformer Encoder Layers, both inclusive and exclusive, can be seen in Table 1. The clustered information gives the model more features to learn from and increases performance in the validation set. Though the performance in the test set is not affected as much. This is because the clustering algorithm here is only trained on the train and validation set and not on the test set which affects the clusters on new data points (test set). Adding BiLSTM over the transformer encoder slightly affected the performance with an increase in the metrics for validation and a slight decrease in the test set. Most of the performance gain comes from the transformer encoder layer which helps the chunk [CLS] embeddings to attend to each other giving the overall document representation, while the cluster labels provide a few extra hidden features to improve the performance slightly. The footnote[2] contains the code used for these experiments.

4 CONCLUSION

This work introduces a framework to classify large unstructured legal documents using both a supervised and unsupervised learning mechanism achieving higher metrics on the experiments on the ILDC dataset over the previous baseline architectures. We demonstrate the effect of including features generated from an unsupervised clustering mechanism and see some relative gain with the same. We aim to explore further to extract the explanation of these predictions in the future and also develop methods to learn from long sequences.

ACKNOWLEDGMENTS

This work is supported by LAWBOT project (ANR-20-CE38-0013) and HPC/AI resources from GENCI-IDRIS (2022-AD011013937).

REFERENCES

[1] Ilias Chalkidis, Manos Fergadiotis, Prodromos Malakasiotis, Nikolaos Aletras, and Ion Androutsopoulos. 2020. LEGAL-BERT: The Muppets straight out of Law School. *CoRR* abs/2010.02559 (2020).

[2] Vijit Malik, Rishabh Sanjay, Shubham Kumar Nigam, Kripa Ghosh, Shouvik Kumar Guha, Arnab Bhattacharya, and Ashutosh Modi. 2021. ILDC for CJPE: Indian Legal Documents Corpus for Court JudgmentPrediction and Explanation. *CoRR* abs/2105.13562 (2021). arXiv:2105.13562 https://arxiv.org/abs/2105.13562

[3] Leland McInnes, John Healy, and Steve Astels. 2017. hdbscan: Hierarchical density based clustering. *Journal of Open Source Software* 2, 11 (2017), 205. https://doi.org/10.21105/joss.00205

[4] Raghavendra Pappagari, Piotr Zelasko, Jesús Villalba, Yishay Carmiel, and Najim Dehak. 2019. Hierarchical Transformers for Long Document Classification. In *2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*. 838–844. https://doi.org/10.1109/ASRU46091.2019.9003958

[5] Shounak Paul, Arpan Mandal, Pawan Goyal, and Saptarshi Ghosh. 2022. Pre-training Transformers on Indian Legal Text. https://doi.org/10.48550/ARXIV.2209.06049

[6] Nishchal Prasad, Mohand Boughanem, and Taoufiq Dkaki. 2022. Effect of Hierarchical Domain-specific Language Models and Attention in the Classification of Decisions for Legal Cases. In *Proceedings of the 2nd Joint Conference of the Information Retrieval Communities in Europe (CIRCLE 2022), Samatan, Gers, France, July 4-7, 2022 (CEUR Workshop Proceedings, Vol. 3178)*. CEUR-WS.org.

[2] https://github.com/NishchalPrasad/Semi-supervised-Stacked-Encoder.git

FALQU: Finding Answers to Legal Questions

Behrooz Mansouri
behrooz.mansouri@maine.edu
University of Southern Maine
Portland, Maine, USA

Ricardo Campos
ricardo.campos@ipt.pt
Polytechnic Institute of Tomar - Ci2 - Smart Cities
Research Center / INESC TEC. Portugal

ABSTRACT

This paper presents a new test collection for Legal IR, FALQU: **F**inding **A**nswers to **L**egal **Q**uestions, where questions and answers were obtained from Law Stack Exchange (LawSE), a Q&A website for legal professionals, and others with experience in law. Much in line with Stack overflow, Law Stack Exchange has a variety of questions on different topics such as copyright, intellectual property, and criminal laws, making it an interesting source for dataset construction. Questions are also not limited to one country. Often, users of different nationalities may ask questions about laws in different countries and expertise. Therefore, questions in FALQU represent real-world users' information needs thus helping to avoid lab-generated questions. Answers on the other side are given by experts in the field. FALQU is the first test collection, to the best of our knowledge, to use LawSE, considering more diverse questions than the questions from the standard legal bar and judicial exams. It contains 9880 questions and 34,145 answers to legal questions. Alongside our new test collection, we provide different baseline systems that include traditional information retrieval models such as TF-IDF and BM25, and deep neural network search models. The results obtained from the BM25 model achieved the highest effectiveness.

1 FALQU TEST COLLECTION

Despite being a recent research area, legal information retrieval has been at the forefront of research efforts with the surge of a few QA legal datasets. The most notable, are COLIEE-2015 [4], which uses Japanese Legal Bar exams and JEC-QA [9], which uses questions from the National Judicial Examination of China. Notwithstanding the emergence of these initiatives, datasets still lack diversity in terms of the questions posed and the domains addressed. Platforms such as Stack Exchange have proved to be a good solution to this problem by providing community question-answering networks for different domains. Such networks have been used over the years within the context of Code Summarization [3] and Math Information Retrieval [7] tasks. However, despite their usefulness, community question-answering websites have never been used for legal information retrieval purposes. In this work, Law Stack Exchange[1] (LawSE) is used to build a new test collection for legal information retrieval, a task that is generally understood as the process of finding answers to legal questions or a single answer as is the case in LawSE. This differs from common IR tasks, where the user is usually interested in retrieving the most relevant documents and not a particular one. Such a task can be formally defined as follows: given a legal question, represented by the question's title and body, an IR model should be able to find (search for) and retrieve the relevant answer (qualified as such by the asker) among all the answers (posts) available in the reference dataset. To build our test collection, we used the 08-Oct-2022 snapshot of LawSE obtained from the Internet Archive. [2] Such snapshot contains 24,187 law questions, with 10,129 having an accepted answer (qualified by the asker as a relevant one). As a means to eliminate duplicate questions from the dataset, we resorted to the available duplicate links feature. Duplicate links refer to links that point to the same (or almost similar) question that has already been posted. After this curated process, we end up with a collection made of 9,880 questions with an accepted answer.

To select the questions for the training and test set, we first split the total set of 9880 questions into 10 bins based on the questions' scores.[3] Binning by score can guarantee that questions of training and test set contain questions of similar quality. After binning, from each bin, we randomly (with a fixed seed to guarantee reproducibility) split 90% of questions for the training set (8892 questions) and the remaining 10% for the test set (988 questions). Each set has a TREC-formatted QREL file in a Tab Separated Value (TSV) file with four columns: query-number 0 document-id relevance-score; where query-number is the question id, document-id is the answer id, and relevance-score is always 1. In our setting, there is only one answer in th QREL for each question. Such answer is considered the relevant one (with a relevance score of 1) as it is the accepted answer qualified as such by the asker. Any other answer not found in the QREL file can be considered non-relevant.

After generating the training and test questions, we then compiled all the answers (among all the posts obtained for the 9880 questions), resulting in 34,145 answers. The compiled answers are provided in TREC format, with tags <DOC>, <DOCNO> which is the actual LawSE answer (post) id, and <TEXT>. Questions are provided in XML format with each question having the <ID> tag that is the actual post id on LawSE, plus the <TITLE>, and the <BODY> tags having the actual LawSE title and body of the question with its corresponding text. Figure 1 shows a sample question (upper part of the figure) and a sample answer (bottom) in the FALQU test collection. Both FALQU test collection as well as the code to generate it have been made publicly available on GitHub for research purposes.[4] For ease of use, we have separated the training and test topics files along with their related QRELs.

A brief analysis of the dataset, shows that FALQU questions have a variety of subjects, from simple questions such as "If a malicious website steals my credit card info, what happens?" to more complex ones involving reasoning and historical knowledge, such as "How would the actions of Hänsel and Gretel in the Grimm tale be

[1] https://law.stackexchange.com/

[2] https://archive.org The referred collection is, due to Internet Archive policies, granted for scholarship and research purposes.

[3] This score, which is a feature of LawSE, is computed as the difference between all the positive and negative votes given by all the users (not specifically the asker), ranging from -9 to 226 in this snapshot.

[4] https://github.com/AIIRLab/FALQU

Table 1: Sample Questions with P@1=1 (+Answer) and P@1=0 (-Answer) with BM25 with YAKE. (Questions' titles shown)

Question	Child Arrangements Order non-biological relative living arrangements
+ Answer	This means that you have the right to make arrangements to do things like arrange for the child to travel...
Question	Notice period after tenancy agreement runs out
- Answer	With respect to disciplining its students and employees, a private school can basically do whatever it wants...

```
...
<Question>
  <ID>17243</ID>
  <TITLE>Should a lease letter ...</TITLE>
  <BODY>I am signing a lease ...</BODY>
</Question>
...
```

```
...
<DOC>
  <DOCNO>12</DOCNO>
  <TEXT>Internationally, according to... </TEXT>
</DOC>
...
```

Figure 1: Example question and answer in FALQU test collection files.

Table 2: Mean Reciprocal Rank(MRR) @1000 and Precision@1 for baselines models on test questions.

Model	MRR@1000	P@1
TF-IDF	0.352	0.274
BM25	0.349	0.270
TF-IDF (YAKE)	0.407	0.313
BM25 (YAKE)	**0.414**	**0.323**
distilroberta	0.337	0.243
all-MiniLM-L12-v2	0.372	0.293
distilroberta (Fine-tuned)	0.368	0.283
all-MiniLM-L12-v2 (Fine-tuned)	0.363	0.276

interpreted in modern law?". There are also questions specific to the law of a specific country, such as "As an Iranian, can I sign an Independent contractor agreement, and work remotely for a EU company from Iran?". Note that all the questions and answers are in English.

2 BASELINE MODELS

We provide several baseline systems, including traditional IR models such as BM25 and TF-IDF and two BERT-based models, using Sentence-BERT [8]. For traditional retrieval models, we consider 2 types of queries: (1) Question titles as the query, and (2) Keywords extracted from the question body as a bag of words + the question title.

To extract keywords, we used YAKE [1] keyword extraction algorithm. Top-5 keywords were extracted per question. Then, to compute the similarity between questions and answers, we resort to TF.IDF and BM25 PyTerrier [6] implementation models. For Sentence-BERT, we considered two pre-trained models, 'all-distilroberta-v1' and 'all-MiniLM-L12-v2'. We fine-tuned both models using all the questions in the training set. For each question, the positive pair is the question and its accepted answer, and the negative pair is the question and a random answer (any other answer than the accepted answer) in the collection. We used 100 epochs and split the training data into 90-10 percent training and validation sets. The best parameters minimize the combination of two loss functions, contrastive [5] and multiple negatives ranking [2] loss.

The systems' effectiveness is compared using two measures: Mean Reciprocal Rank (MRR@1000) and P@1. We choose top-1000 per TREC run criteria of retrieving top-1000. These two measures fit the purposes of this task as there is only one relevant answer per question. The final results are shown in Table 2 using macro-average values. As shown, the highest MRR@1000 (per TREC tasks standard) and P@1 are achieved using the BM25 model with YAKE, significantly better than all the other baselines, except for TF-IDF with YAKE, using the student t-test with p-value< 0.05. Looking at BERT-based models, fine-tuning the models could provide a slight improvement for the 'distilroberta' model. Still, both fine-tuned and pre-trained Bert-based models are less effective when compared to the traditional IR models. Table 1 shows two questions for which BM25+YAKE retrieved relevant and non-relevant answers. When looking at the instances where P@1=0, one can observe that baseline models were able to retrieve answers that might be relevant, but they are not considered as the accepted answer or were answers given to other similar questions. This yields the importance of manual annotation of candidate answers, which will be left for future work.

3 CONCLUSION

In this paper, we introduced and made available FALQU (Finding Answers to Legal Questions) a new test collection, which contains 8892 training and 988 test questions along with a relevant answer for each question. Further to this, we have conducted an experiment with different baselines including TF-IDF, BM25, and Sentence-BERT models for this task. To measure effectiveness, we considered two measures: Precision@1 and Mean Reciprocal Rank@1000. BM25 model with question title and keywords from the question body achieved the highest effectiveness, considering both measures. We hope FALQU can be used in the future by researchers in the legal information retrieval field and extend this test collection for further usage than retrieval, such as legal question answering where an answer is generated rather than being retrieved.

ACKNOWLEDGMENTS

Ricardo Campos was financed by National Funds through the FCT - Fundação para a Ciência e a Tecnologia, I.P. (Portuguese Foundation for Science and Technology) within the project StorySense, with reference 2022.09312.PTDC.

REFERENCES

[1] Ricardo Campos, Vítor Mangaravite, Arian Pasquali, Alípio Jorge, Célia Nunes, and Adam Jatowt. 2020. YAKE! Keyword Extraction from Single Documents using Multiple Local Features. *Information Sciences* (2020).

[2] Matthew Henderson, Rami Al-Rfou, Brian Strope, Yun-Hsuan Sung, László Lukács, Ruiqi Guo, Sanjiv Kumar, Balint Miklos, and Ray Kurzweil. 2017. Efficient Natural Language Response Suggestion for Smart Reply. *arXiv preprint arXiv:1705.00652* (2017).

[3] Srinivasan Iyer, Ioannis Konstas, Alvin Cheung, and Luke Zettlemoyer. 2016. Summarizing Source Code Using a Neural Attention Model. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*.

[4] Mi-Young Kim, Randy Goebel, and S Ken. 2015. COLIEE-2015: Evaluation of Legal Question Answering. In *Ninth International Workshop on Juris-informatics (JURISIN 2015)*.

[5] Xuxing Liu, Xiaoqin Tang, and Shanxiong Chen. 2021. Learning a Similarity Metric Discriminatively with Application to Ancient Character Recognition. In *International Conference on Knowledge Science, Engineering and Management*. Springer.

[6] Craig Macdonald and Nicola Tonellotto. 2020. Declarative Experimentation in Information Retrieval using PyTerrier. In *Proceedings of ICTIR 2020*.

[7] Behrooz Mansouri, Vít Novotnỳ, Anurag Agarwal, Douglas W Oard, and Richard Zanibbi. 2022. Overview of ARQMath-3 (2022): Third CLEF Lab on Answer Retrieval for Questions on Math. In *International Conference of the Cross-Language Evaluation Forum for European Languages*. Springer.

[8] Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*.

[9] Haoxi Zhong, Chaojun Xiao, Cunchao Tu, Tianyang Zhang, Zhiyuan Liu, and Maosong Sun. 2020. JEC-QA: a Legal-domain Question Answering Dataset. In *Proceedings of the AAAI Conference on Artificial Intelligence*.

Opening the TAR Black Box: Developing an Interpretable System for eDiscovery Using the Fuzzy ARTMAP Neural Network

Charles Courchaine
National University
United States
charles@courchaine.dev

Ricky J. Sethi
Fitchburg State University
National University
United States
rickys@sethi.org

CCS CONCEPTS

• **Information systems** → *Evaluation of retrieval results*; *Information retrieval.*

KEYWORDS

TAR, Legal, eDiscovery, Fuzzy ARTMAP

ACM Reference Format:
Charles Courchaine and Ricky J. Sethi. 2023. Opening the TAR Black Box: Developing an Interpretable System for eDiscovery Using the Fuzzy ARTMAP Neural Network. In *Proceedings of European Conference on Information Retrieval (ECIR '23).* ACM, New York, NY, USA, 3 pages. https://doi.org/XXXXXXX.XXXXXXX

1 INTRODUCTION

Technology-assisted review (TAR) utilizes an information retrieval system to discover all, or nearly all, the relevant documents in a corpus and help reduce the human effort required to find these documents [7, 9, 20]. TAR systems are employed in high-recall tasks such as e-discovery, systematic literature reviews, evidence-based medicine, and information test collection annotation [9, 20]. These systems often employ a document classifier and an active learning component to select what documents a human should review [8, 21]. A TAR system that can explain how and why document relevance predictions are made is a vital tool for enabling attorneys to meet their ethical obligations to clients and enable clients to fully participate in the process [12]. Despite the benefits of an explainable TAR system, current systems fail to deliver on why documents are classified as responsive and so these systems are still typically perceived as "black boxes" by practitioners [7, 17].

While a few studies have attempted to bring explainability to TAR systems, they focused on extracting snippets from the documents as the mechanism of explanation rather than directly explaining the relevance model [7, 17]. Instead, we looked at the explainable Fuzzy ARTMAP algorithm. The model learned by the Fuzzy ARTMAP algorithm can be directly interpreted geometrically [4, 19] or as a set of fuzzy If-Then rules [5, 6], depending on the features used.

ECIR '23, April 02-06, 2023, Dublin, Irland

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM…$15.00
https://doi.org/XXXXXXX.XXXXXXX

We performed an initial evaluation of the performance of the explainable Fuzzy ARTMAP algorithm in the TAR domain and found robust performance in terms of recall and precision [10]. Building on the strength of these initial results, we have now continued this foundational research by:

- performing a hyperparameter sweep to refine the parameters
- evaluating the system against the 20Newsgroups, Reuters-21578, RCV1-v2, and Jeb Bush emails corpora for recall, precision, and F_1, and
- generating If-Then rules of document relevance

While these corpora are not specific to the legal domain, the RCV1-v2 and Jeb Bush emails corpora are frequently used in e-discovery evaluations [20, 22] because legal matters are often confidential [7, 9] and their corpora are unavailable. The 20Newsgroups corpus is commonly used as a test corpus with ART-based algorithms [18, 19]; it and the Reuters-21578 corpus are also commonly used in evaluating text classification algorithms [1].

2 FUZZY ARTMAP

Adaptive Resonance Theory (ART) describes how the brain learns and predicts in a non-stationary world [14]. This theory models how brains can quickly learn new information without forgetting previously learned information. ART has been implemented in numerous neural network architectures for supervised, unsupervised, and reinforcement learning applications [3]. Fuzzy ART is a neural network algorithmic instantiation of ART that utilizes operators from fuzzy set theory; specifically, the fuzzy AND operator, to work with real-valued features [4]. The supervised version of the Fuzzy ART algorithm is the Fuzzy ARTMAP algorithm that maps between inputs and categories. By integrating fuzzy set theory and ART dynamics in the Fuzzy ARTMAP neural network algorithm, various interpretations of the learned model are possible. What the model learns may then be represented as fuzzy If-Then text-based rules or depicted geometrically [4, 13].

To take advantage of the geometric interpretation, however, the input must be complement encoded. Complement encoding is a normalization method when working with Fuzzy ARTMAP [4] in which the input vector $\boldsymbol{x}$ is concatenated with its complement $\overline{\boldsymbol{x}}$, yielding an input of $\boldsymbol{I} = [\boldsymbol{x}, \overline{\boldsymbol{x}}]$. As a result, the categories learned by the Fuzzy ARTMAP algorithm can be interpreted as n-dimensional hyper-rectangles [4, 19]. When interpreting the model geometrically, the learned weights from the first half of the vector, the non-complement encoded portion, form one corner of the hyper-rectangle, and the second half of the vector, the complement-encoded portion, forms the other corner as illustrated in Figure 1.

3 METHOD

For the 20Newsgroups, Reuters-21578, RCV1-v2, and Jeb Bush emails corpora, we used tf-idf features with the smaller corpora and the 300-dimension versions of the GloVe and Word2Vec vectorizations with all of the corpora. All the topics in 20Newsgroups, 120 topics in Reuters-21578, and 30 topics in both the RCV1-v2 and the Jeb Bush emails corpora were used for evaluation; the RCV1-v2 and the Jeb Bush corpora were down-sampled to 20% and 50% per [22] due to memory constraints, retaining the general prevalence per topic. For each topic, the Fuzzy ARTMAP algorithm was trained with ten relevant documents and 90 non-relevant documents regardless of corpora size, and the review was run with batches of 100 for the smaller corpora and 1,000 for the larger corpora. The review of documents for each topic concluded when the algorithm predicted no more relevant documents in the unevaluated portion of the corpus. The Fuzzy ARTMAP algorithm was modified to report the degree of fuzzy subsethood [4, 16] associated with documents predicted as relevant, and this degree of fuzzy subsethood was then used to rank the documents for active learning. Based on the results of a sweep of the Fuzzy ARTMAP neural network algorithm hyperparameters, which evaluated different combinations of vigilance (ρ) and learning rates (β), vigilance was set to .95, and a fast learning rate of 1.0 was selected.

A proof-of-concept of one of these If-Then rules for the tf-idf vectorization was produced for predicting documents belonging to the pc.hardware category of the 20Newsgroups dataset, reproduced in Table 1. The tf-idf feature is in italics, and the level of prevalence is in bold. For this example, the level of prevalence was quantized into three levels: rarely, somewhat, and highly prevalent. Additionally, an example of the geometric interpretation is shown and discussed in Figure 1.

4 RESULTS AND DISCUSSION

Considering all corpora and vectorizations, the Fuzzy ARTMAP-based system achieved 100% recall 31% of the time, and achieved the suggested floor of 75% [15] or better recall 67% of the time, as seen

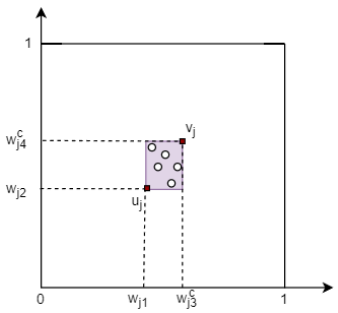


Figure 1: With complement encoding and a 2-dimensional input, the j^{th} category represented by weight vector w can be interpreted geometrically as a rectangle with corners $\mathbf{u}_j$ and $\mathbf{v}_j$, with $\mathbf{u}_j$ corresponding to the first and second positions of the vector, and $\mathbf{v}_j$ corresponding to the complement encoded third and fourth positions. The circles inside the rectangle indicate inputs that fall within the category bounds.

Table 1: Excerpt of Rule Output for pc.hardware

Document is Relevant	
IF	*advance* is **rarely** prevalent in document
and	*apr* is **rarely** prevalent in document
and	*bogus* is **rarely** prevalent in document
and	*browning* is **highly** prevalent in document
and	*calstate* is **rarely** prevalent in document
and	*drive* is **somewhat** prevalent in document
...	

for median recall, precision, and F_1 in Table 2. Recall between the vectorizers for the Reuters-21578 and 20Newsgroups corpora was different by a statistically significant degree based on a Friedman test [11] with $p < .001$ ($\chi^3(2)=25.09$ and $\chi^3(2)=34.9$). A post-hoc Nemenyi test [11] indicated a difference between tf-idf and both GloVe and Word2Vec, with the average difference and statistical significance shown in Table 3. Based on the average difference, there is a practical significance to the tf-idf vectorization over GloVe and Word2Vec. No statistical or practical difference was present between GloVe and Word2Vec for the RCV1-v2 or Jeb Bush Emails corpora.

These results indicate generally robust recall performance, particularly with the tf-idf vectorization. Except for the Jeb Bush Emails, and the GloVe vectorization of 20Newsgroups, the median recall was 75% or better. In the more informal corpora of 20Newsgroups and the Jeb Bush Emails, the GloVe and Word2Vec features did not perform as well. However, this may be due to the corpus specificity of tf-idf compared with the off-the-shelf vocabulary of GloVe and Word2Vec. This suggests that generating corpus-specific GloVe and Word2Vec representations may perform better than the default vocabulary. Future research opportunities exist in optimizing the If-Then rule generation for the tf-idf vectorization and presenting textual and graphical explanations of Word2Vec and GloVe vectorizations. Additionally, exploring corpus-specific versions of Word2Vec and GloVe may bring recall in line with tf-idf, presenting a more efficient yet equally robust option.

While If-Then rules and graphical representations are acknowledged methods of explainability, there are no agreed-upon quantitative metrics for the explainable artificial intelligence space generally [2]; in addition, there are also no qualitative or quantitative user studies of the existing prior attempts at explainability in e-discovery TAR [7, 17]. Therefore, this represents another likely productive area of future work.

Conclusion: This foundational research provides additional support for using the Fuzzy ARTMAP neural network as a classification algorithm in the TAR domain. While research opportunities exist to improve recall performance and explanation, the robust recall results from this study and the proof-of-concept demonstration of If-Then rules for tf-idf vectorization strongly substantiate that a Fuzzy ARTMAP-based TAR system is a potentially viable explainable alternative to "black box" TAR systems.

REFERENCES

[1] Berna Altınel and Murat Can Ganiz. 2018. Semantic Text Classification: A Survey of Past and Recent Advances. *Information Processing & Management* 54, 6 (Nov. 2018), 1129–1153. https://doi.org/10.1016/j.ipm.2018.08.001

Table 2: Median Metrics by Corpus-Vectorizer

Corpus	Vectorizer	Recall	Precision	F_1
20 Newsgroups	GloVe	0.57	0.522	0.434
	Word2Vec	0.772	0.41	0.523
	tf-idf	0.94	0.367	0.53
Jeb Bush Emails	GloVe	0.622	0.07	0.125
	Word2Vec	0.593	0.055	0.098
RCV1-v2	GloVe	0.764	0.211	0.324
	Word2Vec	0.752	0.187	0.292
Reuters-21578	GloVe	0.909	0.384	0.526
	Word2Vec	0.931	0.514	0.624
	tf-idf	0.92	0.733	0.759

Table 3: Average Recall Difference

	Reuters-21578	20Newsgroups
tf-idf-GloVe	0.085^{**}	0.451^{**}
tf-idf-Word2Vec	0.069^{*}	0.171^{**}

$^{*}p < .05$, $^{**}p < .01$

[2] Alejandro Barredo Arrieta, Natalia Díaz-Rodríguez, Javier Del Ser, Adrien Bennetot, Siham Tabik, Alberto Barbado, Salvador Garcia, Sergio Gil-Lopez, Daniel Molina, Richard Benjamins, Raja Chatila, and Francisco Herrera. 2020. Explainable Artificial Intelligence (XAI): Concepts, Taxonomies, Opportunities and Challenges toward Responsible AI. *Information Fusion* 58 (June 2020), 82–115. https://doi.org/10.1016/j.inffus.2019.12.012

[3] Leonardo Enzo Brito da Silva, Islam Elnabarawy, and Donald C. Wunsch. 2019. A Survey of Adaptive Resonance Theory Neural Network Models for Engineering Applications. *Neural Networks* 120 (Dec. 2019), 167–203. https://doi.org/10.1016/j.neunet.2019.09.012

[4] Gail A. Carpenter, Stephen Grossberg, Natalya Markuzon, John H. Reynolds, and David B. Rosen. Sept./1992. Fuzzy ARTMAP: A Neural Network Architecture for Incremental Supervised Learning of Analog Multidimensional Maps. *IEEE Transactions on Neural Networks* 3, 5 (Sept./1992), 698–713. https://doi.org/10.1109/72.159059

[5] Gail A Carpenter and Ah-Hwee Tan. 1993. Rule Extraction, Fuzzy ARTMAP, and Medical Databases. In *Proceedings of the World Congress on Neural Networks*. Erlbaum Associates, Portland, OR, USA, 501–506.

[6] Gail A. Carpenter and Ah-Hwee Tan. 1995. Rule Extraction: From Neural Architecture to Symbolic Representation. *Connection Science* 7, 1 (Jan. 1995), 3–27. https://doi.org/10.1080/09540099508915655

[7] Rishi Chhatwal, Peter Gronvall, Nathaniel Huber-Fliflet, Robert Keeling, Jianping Zhang, and Haozhen Zhao. 2018. Explainable Text Classification in Legal Document Review a Case Study of Explainable Predictive Coding. In *2018 IEEE International Conference on Big Data (Big Data)*. IEEE, Seattle, WA, USA, 1905–1911. https://doi.org/10.1109/BigData.2018.8622073

[8] Rishi Chhatwal, Nathaniel Huber-Fliflet, Robert Keeling, Jianping Zhang, and Haozhen Zhao. 2017. Empirical Evaluations of Active Learning Strategies in Legal Document Review. In *2017 IEEE International Conference on Big Data (Big Data)*. IEEE, Boston, MA, 1428–1437. https://doi.org/10.1109/BigData.2017.8258076

[9] Gordon V. Cormack and Maura R. Grossman. 2015. Autonomy and Reliability of Continuous Active Learning for Technology-Assisted Review. *arXiv:1504.06868 [cs]* (April 2015). arXiv:1504.06868 [cs]

[10] Charles Courchaine and Ricky Sethi, J. 2022. Fuzzy Law: Towards Creating a Novel Explainable Technology-Assisted Review System for e-Discovery. In *2022 IEEE International Conference on Big Data (Big Data)*. IEEE, Osaka, Japan, 1218–1223. https://doi.org/10.1109/BigData55660.2022.10020503

[11] Janez Demsar. 2006. Statistical Comparisons of Classifiers over Multiple Data Sets. *The Journal of Machine Learning Research* 7 (Dec. 2006), 1–30.

[12] Seth Katsuya Endo. 2018. Technological Opacity & Procedural Injustice. *Boston College Law Review* 59, 3 (March 2018), 822–875. https://doi.org/bclr/vol59/iss3/2

[13] Stephen Grossberg. 2020. A Path Toward Explainable AI and Autonomous Adaptive Intelligence: Deep Learning, Adaptive Resonance, and Models of Perception, Emotion, and Action. *Frontiers in Neurorobotics* 14 (June 2020), 36. https://doi.org/10.3389/fnbot.2020.00036

[14] Stephen Grossberg. 2021. Toward Autonomous Adaptive Intelligence: Building Upon Neural Models of How Brains Make Minds. *IEEE Transactions on Systems, Man, and Cybernetics: Systems* 51, 1 (Jan. 2021), 51–75. https://doi.org/10.1109/TSMC.2020.3041476

[15] Robert Keeling, Rishi Chhatwal, Peter Gronvall, and Nathaniel Huber-Fliflet. 2020. Humans Against the Machines: Reaffirming the Superiority of Human Attorneys in Legal Document Review and Examining the Limitations of Algorithmic Approaches to Discovery. *Richmond Journal of Law & Technology* 26, 3 (2020), 65.

[16] Bart Kosko. 1986. Fuzzy Entropy and Conditioning. *Information Sciences* 40, 2 (Dec. 1986), 165–174. https://doi.org/10.1016/0020-0255(86)90006-X

[17] Christian J. Mahoney, Jianping Zhang, Nathaniel Huber-Fliflet, Peter Gronvall, and Haozhen Zhao. 2019. A Framework for Explainable Text Classification in Legal Document Review. In *2019 IEEE International Conference on Big Data (Big Data)*. IEEE, Los Angeles, CA, USA, 1858–1867. https://doi.org/10.1109/BigData47090.2019.9005659

[18] Dušan Marček and Michal Rojček. 2017. The Category Proliferation Problem in ART Neural Networks. *Acta Polytechnica Hungarica* 14, 5 (2017), 15.

[19] Lei Meng, Ah-Hwee Tan, and Donald C. Wunsch II. 2019. *Adaptive Resonance Theory (ART) for Social Media Analytics*. Springer International Publishing, Cham, 45–89. https://doi.org/10.1007/978-3-030-02985-2_3

[20] Eugene Yang, David D. Lewis, and Ophir Frieder. 2019. A Regularization Approach to Combining Keywords and Training Data in Technology-Assisted Review. In *Proceedings of the Seventeenth International Conference on Artificial Intelligence and Law*. ACM, Montreal QC Canada, 153–162. https://doi.org/10.1145/3322640.3326713

[21] Eugene Yang, David D. Lewis, and Ophir Frieder. 2021. On Minimizing Cost in Legal Document Review Workflows. *arXiv:2106.09866 [cs]* (June 2021). https://doi.org/10.1145/3469096.3469872 arXiv:2106.09866 [cs]

[22] Eugene Yang, Sean MacAvaney, David D. Lewis, and Ophir Frieder. 2021. Goldilocks: Just-right Tuning of BERT for Technology-Assisted Review. *arXiv:2105.01044 [cs]* (May 2021). arXiv:2105.01044 [cs]

Parsing User Queries using Context Free Grammars

Kees van Noortwijk
vannoortwijk@law.eur.nl
Erasmus School of Law
Rotterdam, The Netherlands
Rechtsorde BV
Den Haag, The Netherlands

Christian F. Hirche
hirche@me.com
Rechtsorde BV
Den Haag, The Netherlands

ABSTRACT

In legal information retrieval, query cooking can significantly improve recall and precision. Context free grammars can be used to effectively parse user queries, even if the number of items to recognize is high and recognition patterns are complicated.

CCS CONCEPTS

• **Information systems** → **Query intent**; *Link and co-citation analysis*; • **Applied computing** → *Law*.

KEYWORDS

legal information retrieval, query cooking, text parsing, context free grammars

ACM Reference Format:
Kees van Noortwijk and Christian F. Hirche. 2023. Parsing User Queries using Context Free Grammars. In *Proceedings of The first international workshop on Legal Information Retrieval, to be held at ECIR 2023 (LegalIR '23).* ACM, New York, NY, USA, 4 pages.

1 INTRODUCTION

The use of digital information sources these days is a vital part of the work of almost every lawyer, now that traditional information sources such as books and journals to a large extent have been replaced by their digital counterparts.[1] The retrieval systems used to search these digital collections and retrieve relevant legal documents usually have access to millions of documents. Because of that, even basic queries consisting of just one or two keywords usually deliver a few relevant documents, be it as part of a much larger set of not-so-very-relevant ones. However, that is often not enough for professional users, who not only want information that is as *complete* as possible, but who also do not want to wade through large amounts of irrelevant stuff to eventually find what they are looking for. In other words, a legal information retrieval system should be finetuned to deliver optimal recall and precision, with results carefully ranked according to their relevancy. In [5] it is argued that specifically recall – the ratio between the number of relevant documents retrieved and the number of such documents being present in the database – is important from the legal perspective, but is often also difficult to measure.

LegalIR '23, April 2, 2023, Dublin, Republic of Ireland

To optimize recall, it is important that the *initial* query places all documents that could possibly be relevant in the initial list of search results. This list can subsequently be filtered, using 'facets' like the type of document, the area of law, etcetera, to increase precision. But documents absent from that initial list will not be part of the final set, no matter how exact the filtering options will be set. That is why it is important to optimize the results of the initial query: what is missed there, cannot be regained in subsequent (filtering) steps. One way to improve the quality of the initial query is to not take the terms in that query for granted, but to use algorithms to find out what these terms might *mean* and what the intention of the user might be to use them in the query. For instance, if the user would have entered a number followed by the full name or abbreviation for a certain piece of legislation and the words 'case law', it will probably not be useful to just return documents containing this combination of words/items. Instead, the system should look for case law documents containing decisions relating to the article of law that can be derived from the number and the law name. The latter information could be present in the 'body text' of a (case law) document, but also in metadata that are part of it.

This is only one example of a possible improvement of query effectiveness, achieved through analysis – followed by automatic adjustment – of a user query before that query is executed. Another example might be the automatic addition of synonyms to a search query, or the recognition of well-known legal terms to add corresponding articles of law or even certain case law identifiers to the query. This process of analysing and adjusting a query is called *query cooking*. It is probably used in the majority of document retrieval systems these days, but arguably is particularly useful in collections of documents all relating to a particular field or subject area, because in that case algorithms and rules can be applied that relate to that particular subject area. For instance, in the field of law, rules can be defined that are capable of recognising articles of law, case law identifiers or 'nicknames' that might be in use to refer to these, as well as references to legal textbooks and law guides. This paper assesses methods to implement such rules in a retrieval system for various types of legal content, paying attention to functionality as well as to maintainability.

2 QUERY COOKING

A query cooking function in a document retrieval system can perform several functions, such as:

- pattern matching, to find terms or groups of terms that conform to a certain specification; for instance: a number in a particular format, such as the Celex numbers that are used to identify EU documents[2], or a number preceded or followed

by a non-numeric string, which combination could designate a particular law article;
- word group identification, to automatically search sets of terms that constitute one concept as a 'phrase' (as if it would have been enclosed in double quotes);
- identification of known (legal) concepts, nicknames and other keywords, which can be searched by adding to the query corresponding (case law) identifiers, articles of law or other references.

A common characteristic of these functions is that known terms (from previously compiled lists) and patterns need to be identified within the query. Specifically for identifying patterns, regular expressions [4] are often used. A simple regular expression to recognise a Celex-number could for instance be:

```
[0-9cCeE]\d{4}\D{1,2}\d{3,4}.*
```

Law articles are already more complex to cover, as they consist of at least two elements (the law abbreviation, to be matched against a list, and the article number). However, as Van Opijnen et al. ([6], par. 3.4) already stated, regular expressions can have drawbacks in large-scale environments, as multiple types of items to recognize and many possible matches can lead to very complicated setups that can be difficult to debug and maintain. Instead, they proposed an alternative for the specific task of recognising legal references in document texts, in the form of *grammars*, in particular so-called Parsing Expression Grammars (PEGs). A grammar is a set of rules used to recognize language elements. In the case of PEGs, this recognition is performed *without ambiguity*, in other words, each string that is parsed can have only one valid 'parsing tree' at the most. Any possible choices that might result from the grammar are considered in an ordered form, choosing the first valid option while ignoring subsequent ones. Theoretically, this can be expected to work well for parsing strings containing strictly-defined legal references, as such references can be resolved to one and only one publication.

In practice, however, precluding ambiguity when parsing legal references does not always work well. In some cases, two or more publications can share the same title, abbreviation, or other identifying designation. Then, a legal reference containing such an ambiguous designation can become ambiguous itself. Aggravating the problem, in case of parsing of user queries, legal references are often short and miss context, which makes them more prone to ambiguity.

In addition, even when a legal reference can be parsed unambiguously, its surrounding context, which usually is just natural language content, cannot be parsed unambiguously (see for example [3]). Therefore, when attempting to use PEGs to parse legal references inside a longer text, a two-step approach is necessary. In the first step, unambiguous legal references must be identified and separated from surrounding text. In the second step, the actual parsing will occur.

3 CONTEXT FREE GRAMMARS

As an alternative approach, which does not suffer from these issues, so-called Context Free Grammars (CFGs) can be used. These grammars allow for ambiguity, which means that, in principle, parsing a text could result in several alternative parse trees. Choosing one parse tree over the other is done using priorities assigned to parse rules. First, this makes parsing ambiguous legal references possible. Second, CFGs can also be used to parse the text surrounding a legal reference, which cannot be parsed by a PEG, eliminating the need to use a two-step approach.

In the case of a user search query, ambiguous ways of parsing will lead to alternative interpretations of the query. These alternative interpretations can either be discarded or can be used to create a (processed) query containing elements that are to be searched alternatively (Boolean: OR). Usually, that is exactly what is needed here: queries are seldomly completely exact and can contain combinations of terms of which only a subset is present in the document the user intends to find. Query cooking can help to make the most of what was input, at the same time providing information that can subsequently be used for the optimal ranking of search results – for instance by adding 'boosting' to documents that exactly match recognised elements.

Query parsing using custom-made CFGs is now used in the Dutch legal information retrieval system Rechtsorde. It uses an implementation of an Earley parser. A slightly simplified excerpt of the grammar to identify a reference to the law "Burgerlijk Wetboek" (the Dutch Civil Code) is shown below:

Listing 1: Excerpt of example grammar to recognise legal references

```
text: (legal_reference delimiter |
    any_other_text delimiter | delimiter)*
legal_reference: regular_law | bw |
    publication //...
any_other_text.-100: ANY_CHARACTER //low
    priority to default to a legal reference

bw: [bw_law_prefix SEP] bw_references |
    identifier_bw
bw_references: identifier_bw [SEP]
    bw_book_reference [SEP
    bw_article_reference [SEP
    part_and_sub_ref]]
| bw_book_reference SEP identifier_bw [SEP
    bw_article_reference [SEP
    part_and_sub_ref]]
| bw_book_reference SEP bw_article_reference
     SEP identifier_bw [SEP part_and_sub_ref
    ]
| bw_book_reference SEP bw_article_reference
     SEP part_and_sub_ref SEP identifier_bw
    //...
bw_book_reference: [KEYWORD_BOOK SEP]
    NUM_BOOK
bw_article_reference: [KEYWORD_ARTICLE SEP]
    num_article_bw

KEYWORD_BOOK: "boek"
NUM_BOOK: "1".."8" | "10" | "7a"
//...
```

The grammar in Listing 1 shows the hierarchical construction of a grammar. This means that a starting rule is defined by one or more other rules or terminals, which are defined by one or more other rules or terminals, and so on. The hierarchy ends when a rule is exclusively defined by terminals, which are a character, string, or regular expression. In the example, the starting rule is called `text`. This rule's definition allows for zero or more instances of either (1) a legal reference (rule: `legal_reference`) followed by a delimiter (rule: `delimiter`) or (2) something else (rule: `any_other_text`) followed by a delimiter. One level below, the rule `legal_reference` is defined as the combination of the rules `regular_law`, `bw`, `publication`, and others not shown in the excerpt. The rule `any_other_text`, on the other hand, refers to the terminal `ANY_CHARACTER`, which might be any character or string. The text "-100" behind the rule name indicates that this rule has a low priority and whenever a rule with a higher priority can match, it will have preference over this rule.

Using this grammar to parse a short example text *reference to BW Boek 7* results in a parse tree shown on in Figure 1 on page 4. The parser matches *reference* as rule `any_other_text`, the space character after that as `delimiter`, *to* as rule `any_other_text`, and the space after that again as `delimiter`. Subsequently, *BW* fits the definition of the rule `identifier_bw` (not shown in the excerpt), the space fits the terminal definition for `SEP` and *Boek 7* fits the definition of `bw_book_reference`. Taken together, `identifier_bw`, `SEP`, and `bw_book_reference` fit the definition of `bw_references`, which, in turn, fits the definition of rule `bw`. Several other parse trees are also possible for that text, but have been discarded based on the low priority of the `any_other_text`-rule. Based on this tree, the query cooking process can create a directed query for a legal reference, which is optimised for the field and the format in which these references appear in document metadata.

Implementing this grammar-based form of query cooking has not only improved the overall reliability and speed of the recognition of query elements, but has also made it possible to, in principle, recognise an unlimited number of elements in any single query and process the results of that accordingly. At the same time, maintainability has greatly improved. This result would not have been possible using regular expressions or similar programming techniques.

REFERENCES

[1] David W. Dunlap. 2022. So Little Paper to Chase in a Law Firm's New Library. *New York Times* (October 2022).

[2] EU Publication Office 2000. *Celex Numbers.* Retrieved January 20, 2023 from https://eur-lex.europa.eu/content/help/eurlex-content/celex-number.html

[3] Bryan Ford. 2004. Parsing expression grammars: a recognition-based syntactic foundation. In *Proceedings of the 31st ACM SIGPLAN-SIGACT symposium on Principles of programming languages*. 111–122.

[4] Walter L. Johnson, James H. Porter, Stephanie I. Ackley, and Douglas T. Ross. 1968. Automatic generation of efficient lexical processors using finite state techniques. *Commun. ACM* 11, 12 (December 1968), 805–813.

[5] Kees van Noortwijk. 2017. Integrated Legal Information Retrieval; new developments and educational challenges. *European Journal of Law and Technology* 8, 1 (2017), 1–18.

[6] Marc van Opijnen, Nico Verwer, and Jan Meijer. 2015. Beyond the Experiment: The Extendable Legal Link Extractor. In *Workshop on Automated Detection, Extraction and Analysis of Semantic Information in Legal Texts, held in conjunction with the 2015 International Conference on Artificial Intelligence and Law (ICAIL), June 08 - 12, 2015, San Diego, CA, USA.*

Received 20 January 2023; accepted 3 March 2023

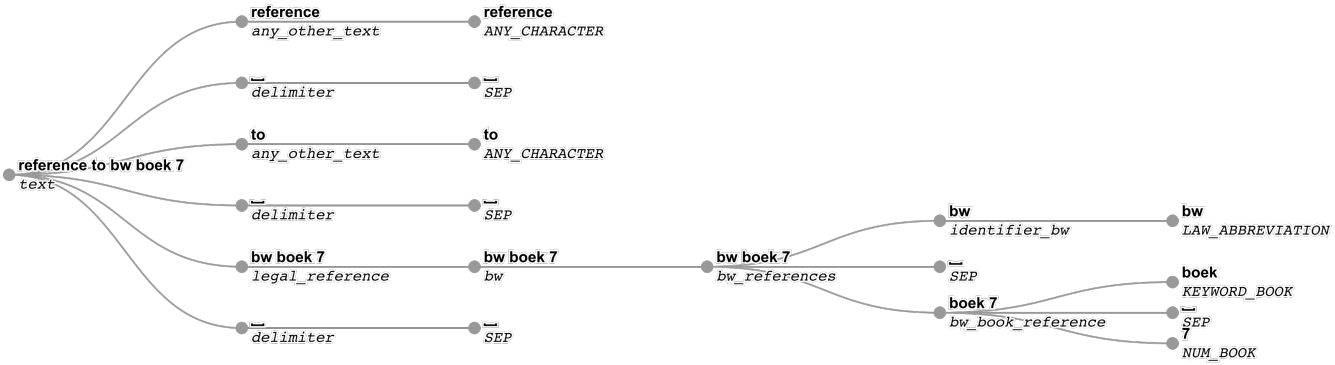


Figure 1: Parse tree for query string "reference to bw boek 7"

Implicit Assumptions in the Evaluation of One-Phase Technology-Assisted Review

David D. Lewis
ECIRLegalIR2023@incized.com
Redgrave Data
Chantilly, VA, USA

ABSTRACT

One-phase technology-assisted review (TAR) has become the dominant TAR approach in eDiscovery. This is despite substantial confusion about how to evaluate these reviews, and even about whether proposed evaluation methods are statistically valid. This confusion results in part from leaving implicit the assumptions required to define the effectiveness of a manual review that is itself evaluated against manual review decisions. We discuss three of these assumptions, and what they imply for evaluation.

CCS CONCEPTS

• **Information systems** → **Information retrieval**; Enterprise applications; *Retrieval effectiveness*; • **Theory of computation** → **Active learning**.

KEYWORDS

AI and law, high recall retrieval, sampling, statistical quality control, total recall, continuous active learning (TM), CAL (TM)

ACM Reference Format:
David D. Lewis. 2023. Implicit Assumptions in the Evaluation of One-Phase Technology-Assisted Review. In *Proceedings of (ALTARS 2023).* ACM, New York, NY, USA, 2 pages. https://doi.org/XXXXXXX.XXXXXXX

1 INTRODUCTION

One-phase (aka continuous or review-oriented) technology-assisted review (TAR) workflows [5] use iterative active learning and other technologies to prioritize documents for manual review. In contrast to two-phase (classifier-oriented) TAR workflows [5], the evaluation focuses on review decisions, not the behavior of individual text classifiers.

One-phase workflows are increasingly dominant for document review in electronic discovery (eDiscovery) in the law, as well as seeing use in systematic literature reviews. Curiously, no clearly accepted approach to evaluating one-phase reviews has emerged in operational practice. This is in contrast to two-phase reviews, where estimation of text classification metrics based on a simple random sample from the document collection is widely accepted and implemented in commercial eDiscovery software.

ALTARS 2023, ,

https://doi.org/XXXXXXX.XXXXXXX

One reason for this odd state of affairs is that one can only evaluate a review by a review. Breaking this circularity requires making assumptions, and these assumptions have rarely been made explicit in discussions of one-phase review evaluation. We discuss three such assumptions and their implications for evaluation.

2 ASSUMPTION 1: HANDLING OF UNCODED DOCUMENTS

At any point during a one-phase TAR review, each document in the collection is in one of three states:

- Coded as relevant
- Coded as nonrelevant
- Uncoded

So, just as when evaluating a binary classifier that is allowed a reject output, one must decide how to treat documents which have received neither relevant nor nonrelevant coding.

Two competing assumptions have been made in discussions of one-phase TAR evaluation:

- Ignore Uncoded Documents: Only explicit coding decisions are evaluated.
- Treat Uncoded Documents as if Coded as Nonrelevant: Uncoded documents are considered coded as nonrelevant for evaluation purpopses.

Ignoring of uncoded documents has occasionally been proposed as a component of a larger evaluation approach [1]. However, it presents the obvious difficulty that coded documents are not representative of the collection, and it is effectiveness on the collection that we care about.

The second assumption is the more common one in evaluating one-phase TAR. It acknowledges that uncoded documents will, at the conclusion of review, typically be treated the same way as documents coded as nonrelevant (e.g., not produced to other parties in the legal matter, not used in a systematic review, etc.)

3 ASSUMPTION 2: REVIEW STANDARD

If a manual review is itself evaluated against manual review decisions, does this mean the review itself is treated as correct? One of two contrasting assumptions have been made here:

- *Imperfect Review*: Relevant and nonrelevant coding decisions in the one-phase review under evaluation may be incorrect when judged against some other gold standard review.
- *Perfect Review*: Relevant and nonrelevant coding decisions made during the one-phase review under evaluation are assumed correct. The only potentially incorrect decisions are implicit decisions of nonrelevance implied by the noncoding of documents.

The Imperfect Review assumption might seem natural, given that human review is indeed imperfect. Imperfection is certainly assumed when evaluating individual manual reviewers against a gold standard.

However, most evaluations of one-phase TAR implicitly make the Perfect Review assumption. This seems correct. While review decisions are subjective and errorful, they must be treated as definitive at some point. In eDiscovery, an attorney must sign off on a production being final, with the understanding that the review decisions made were reasonable, not perfect. An estimate of a completed review's effectiveness relative to a hypothetical gold standard is a poor conceptual fit to this scenario. Similarly, in systematic review, at some point search ends and a review is written.

There would be practical problems if the Imperfect Review assumption was adopted in eDiscovery practice. Parties in legal cases may find it helpful to agree on a target value for estimated recall. If the Imperfect Review assumption were made and the review judged against a gold standard sample, it is possible that a producing party could review every document, but still have estimated recall fall below the agreed target.

3.1 Assumption 3: Treatment of Random Sample Coding In Defining Effectiveness

Most evaluation methods are based on sample-based statistical inferences about the effectiveness of review. These require that a set of documents randomly selected from the collection be coded as relevant or nonrelevant, just as in the review being evaluated.

Does this review of the random sample "count" as part of the review that it itself is used to evaluate? From a practical standpoint, the answer is surely yes. If a relevant medical abstract is found during random sampling to evaluate a one-phase systematic literature review, it will be included in the writeup of the review. In eDiscovery, a producing party's responsibilities are identical with respect to the two types of review: responsive documents found when reviewing a random sample must be produced to a requesting party just like responsive documents found during ordinary review.

However, in both the research literature on one-phase TAR evaluation, and in practice, there has been little consistency on this point. Any of three implicit assumptions has been made:

- Sample Coding Ignored: Only coding decisions viewed as resulting from "normal" review are counted when defining the effectiveness of the review. Coding decisions viewed as resulting from coding of the random sample are ignored.
- Sample Coding Counted: All coding decisions are treated equally when defining the effectiveness of a review.
- Inconsistent: Coding of random sample documents is counted for some purposes but not for others. It is usually unclear whether this is a deliberate decision or a confusion resulting from not making assumptions clear.

Ignoring the coding of random sample data is sometimes justified on the grounds that few relevant documents are found this way versus via prioritized review. There are at least four problems with this perspective:

- It is self-fulfilling: if random samples are not counted toward effectiveness, there is strong motivation to keep random samples small, regardless of the downsides of this for evaluation.
- If a review is struggling to meet a recall target, each additional relevant document comes at an increasing cost in reviewing nonrelevant documents. While random samples find fewer relevant documents than prioritized methods, each such document that is credited toward a recall target saves the cost of examining many nonrelevant documents.
- Ignoring random sample coding in defining effectiveness leads to nonsensical results. For instance, suppose TAR has not been used, but a large random sample has been reviewed and found many relevant documents. If coding of random sample documents is ignored, recall is defined to be exactly zero in this situation.
- Ignoring the coding of random sample documents complicates the evaluation of real-world TAR workflows. In operational settings it may be unclear whether a document was reviewed due to "normal" review or random sampling. At a minimum, keeping track of the difference is a distraction from the difficult process of managing a review.

Note that we are considering how effectiveness is defined, not how the defined effectiveness is estimated from the random sample. Estimators of effectiveness also vary in how the coding of random sample documents versus other documents is treated, but the question then is simply whether that results in a good estimator. The more fundamental issue is how one defines the population quantity to be estimated.

4 TAR FOR SMART PEOPLE 3.14159...

In the proposed talk, I will discuss the versions of these assumptions implicit in two published discussions of one-phase TAR evaluation. The first is the book TAR For Smart People, Third Edition [4] distributed by a major eDiscovery company. It includes worked out examples of a procedure widely but nervously used in operational one-phase workflows: estimating recall by sampling only from unreviewed documents. The above three assumptions are critical to their calculations, but are left implicit and are applied inconsistently.

The second is the paper "Certifying One-Phase Technology-Assisted Reviews" by myself, Eugene Yang, and Ophir Frieder [3]. This paper introduces a quantile estimation method for recall which generalizes Cormack & Grossman's Target method [2]. It provides the first general purpose approach to one-phase TAR evaluation that avoids sequential bias. However, the above three assumptions again are left implicit, and when made explicit suggest difficulties with using the method in practical settings. This also suggests some directions for more practical evaluation methods.

REFERENCES

[1] Max W Callaghan and Finn Müller-Hansen. 2020. Statistical stopping criteria for automated screening in systematic reviews. *Systematic Reviews* 9, 1 (2020), 1–14.
[2] Gordon V. Cormack and Maura R. Grossman. 2016. Engineering Quality and Reliability in Technology-Assisted Review. In *SIGIR*. ACM Press, Pisa, Italy, 75–84. https://doi.org/10.1145/2911451.2911510 00024.
[3] David D Lewis, Eugene Yang, and Ophir Frieder. 2021. Certifying one-phase technology-assisted reviews. In *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*. 893–902.
[4] J. Tredennick, J. Pickens, T. Gricks III, and A. Bye. 2018. *TAR for Smart People: How Technology Assisted Review Works and why it Matters for Legal Professionals* (third ed.). Catalyst.
[5] Eugene Yang, David D. Lewis, and Ophir Frieder. 2021. On Minimizing Cost in Legal Document Review Workflows. In *Proceedings of the 21st ACM Symposium on Document Engineering*.