Overview of the FIRE 2020 AILA Track: Artificial Intelligence for Legal Assistance

Paheli Bhattacharya[1], Parth Mehta[2], Kripabandhu Ghosh[3], Saptarshi Ghosh[1], Arindam Pal[4], Arnab Bhattacharya[5], and Prasenjit Majumder[6]

[1] Indian Institute of Technology Kharagpur
[2] Parmonic AI
[3] Indian Institute of Science Education and Research (IISER), Kolkata
[4] Data61, CSIRO, Sydney, Australia
[5] Indian Institute of Technology Kanpur
[6] DA-IICT Gandhinagar

Abstract. The FIRE 2020 AILA track focused on two tasks – (i) Retrieving relevant Prior cases and Statutes given a factual description, and (ii) Rhetorical labelling of sentences in a legal case document, where the rhetorical roles are – Facts of the case, Ruling by the Lower Court, Argument, Statute, Precedent, Ratio of the decision and Ruling by the Present Court. Both the tasks were based documents from the Indian Supreme Court judiciary, which are publicly available.

Keywords: Legal data analytics · Prior case retrieval · Statute retrieval · Legal facts · Rhetorical role labelling · Legal IR

1 Introduction

Common Law system, which is followed by most countries (UK,USA, Canada, Australia, India etc.) has two primary sources – Precedents and Statutes. Precedents are the prior-cases decided in the Courts of law. Statutes are bodies of written law.

A case is presented to a lawyer in terms of a factual description. To prepare legal reasonings accordingly, the law expert has to search for relevant Precedent cases and Statutes. Since these documents are large in number, it will be beneficial for a law practioneer to have an automated tool that can return prior-cases and statutes relevant to a factual scenario. Motivated by this, we design Task 1: Precedent and Statute Retrieval for a given factual description. We had the same task in AILA-2019 [4].

Also, a case document from the Indian judiciary is usually very long and unstructured, lacking section and paragraph headings. This makes it difficult for a reader of the document to identify where the facts of the case are written, which sentences mention the arguments made in the case, what was the final judgement and so on. To address this research problem, we introduce Task 2 : Rhetorical Role Labeling for Legal Judgements.

We describe the tasks below.

1.1 Task 1: Precedent and Statute Retrieval

In this task, the aim is to retrieve relevant prior-cases and statutes for a given factual scenario. Details of how the task is in [4]. For training, we provide the dataset of AILA-2019 [4]. For the test/evaluation data, we provide a set of additional 10 queries, that describe factual scenarios in natural English language. The pool / candidate statutes from which the relevant ones were to be retrieved, was the same as that in AILA-2019 dataset [4]. A set of 343 documents were added to the existing pool of prior-case documents in AILA-2019 [4]. Out of these 343 documents, 43 documents were the relevant prior cases for the 10 queries in the test set. The remaining 300 documents were sampled based on the text similarity (cosine similarity between the document vectors were in $[0.3, 0.5]$) between the existing documents in the prior-case pool.

For each query, the task was to retrieve the most similar/relevant precedent case documents and statutes with respect to the situation in the given query.

Similar research has been done on Chinese legal case documents [14, 13] – the dealt with the task of retrieving statutes for a given fact. Note that, in addition to retrieving statutes, we also consider retrieving prior-cases for the query. Also, constructing a large dataset for the task to train supervised models as in [14, 13] is not practical in the context of Indian legal documents. This is because, unlike Chinese legal documents, Indian legal documents are not well-structured and it is difficult to extract the facts of the cases automatically [5] for creating the dataset.

1.2 Task 2: Rhetorical Role Labeling for Legal Judgements

Since Indian legal case documents are unstructured it is a need of the hour to design systems that can automatically segment these documents into coherent, meaningful parts. This can not only enhance the readability of the documents but also has applications in downstream tasks like summarization, case-law analysis, semantic search and so on. We introduce the task of rhetorical role labeling of sentences in legal case judgements in AILA-2020. The task is to assign one of the following labels to each sentence in a legal case document. We consider the following seven (07) rhetorical labels/semantic segments [6]:

– Facts : legal situation that led to filing the case
– Ruling by Lower Court : since we consider documents from the Supreme Court of India, there was some preliminary ruling given at the lower courts e.g.. High Court, Tribunal etc.
– Argument : arguments made by the contending parties
– Precedents : citation to relevant prior cases

- Statutes : citation to relevant statutes
- Ratio of the decision : reasoning behind the final judgement
- Ruling by Present Court : final judgement given by the Supreme Court of India

A state-of-the-art paper that addresses this task for Indian legal documents is [6]. They use deep learning models and perform 5-fold cross-validation.

2 Dataset

We consider case documents from the Supreme Court of India and Statutes from the Indian judiciary. **Task 1:** The training dataset for Task 1 was the AILA-2019 dataset [4], available at https://github.com/Law-AI/aila-2019-dataset. There were 50 queries and 197 statutes. The pool of prior-cases were extended to having 3257 documents, as mentioned in Section 1.1. The test dataset of 10 queries were created in the same way as in [4].
Task 2: The dataset made publicly available by [6] in https://github.com/Law-AI/semantic-segmentation, was used as the training dataset. There were 50 documents containing $9,308$ sentences in total across all the As the test set, we consider a set of 10 additional case documents. We randomly selected 2 documents from each of the 5 law domains mentioned in [6]. These documents were then given to a law expert for annotating every sentence with one of the rhetorical labels. There are a total of $1,905$ sentences in the test set.

3 Evaluation

For both the Tasks, evaluation was done on the test dataset. For Task 1, the same measures as in [4] were use – Mean Average Precision (MAP), Precision@10 (P10), BPREF and Reciprocal rank (recip_rank) as the evaluation metrics. The *trec_eval* tool [7] tool was used for computing the metrics stated above. We choose MAP as the primary measure since it incorporates both Precision and Recall. For Task 2, we use the standard Recall, Precision and F1-Scores. The documents have a considerable variation in their size. Moreover, even within a document, there is a class imbalance among the 7 categories / rhetorical roles. Hence we use macro-averaging at both document-level and category-level.

4 Methodologies for Task 1: Precedent Retrieval and Statute Retrieval

For the first task of retrieving relevant prior/ precedent cases (Task 1a) , we received a total of 26 runs from 10 participating teams. For the second task of retrieving relevant statutes (Task 1b) , we received a total of 27 runs from 12

[7] https://trec.nist.gov/trec_eval/

Table 1. Results of Task 1a: Precedent retrieval for queries. All measures averaged over 10 test queries. Numbers in **bold** and underline indicate the best and the second-best performing methods corresponding to the evaluation metrics. Rows are sorted in decreasing order of MAP (primary measure).

Team Name	Run_ID	MAP	BPREF	recip_rank	P @ 10	Method Used
UB	UB-3	**0.1573**	**0.1128**	**0.238**	0.08	Terrier 4.2 KL divergence model
double_liu_2020	double_liu_2020_3	0.1382	0.1045	0.1886	0.07	IDF, BM25 as the search score.
fs_hu	fs_hu_task1a	0.1351	0.0885	0.2041	**0.1**	Language model, Dirichlet Smothing
double_liu_2020	double_liu_2020_1	0.1306	0.0737	0.1963	0.07	IDF, BM25
TUW_informatics	basic	0.1294	0.0737	0.1915	0.07	Preprocessing, BM25
fs_hit_1	fs_hit_1_task1a_01	0.1294	0.0877	0.1876	0.07	BM25
LAWNICS	LAWNICS_2	0.1288	0.0913	0.1586	**0.1**	Topic Embedding
TUW_informatics	word_count	0.1271	0.0728	0.1891	0.06	Preprocessing, BM 25
SSNCSE_NLP	task_1a_1	0.1264	0.0918	0.2043	0.08	BM 25
fs_hit_2	fs_hit_2_task1a_01	0.125	0.0724	0.1906	0.07	Language modelling of Indri
double_liu_2020	double_liu_2020_2	0.123	0.0621	0.1969	0.08	All words, BM 25
UB	UB-1	0.1229	0.07	0.2033	0.09	TF-IDF term weighting
fs_hit_1	fs_hit_1_task1a_02	0.1215	0.0699	0.2078	0.09	BM25, TF-IDF
UB	UB-2	0.1168	0.0798	0.1967	0.07	Extract key concepts, TF-IDF
TUW_informatics	false_friends	0.1133	0.0687	0.1873	0.05	Preprocessing, BM25
LAWNICS	LAWNICS_1	0.1085	0.0756	0.1607	0.08	Preprocessing, BM25
Uottawa_NLP	run3_TFIDF	0.0837	0.0399	0.1157	0.05	preprocessing, TFIDF
fs_hit_1	fs_hit_1_task1a_03	0.0696	0.0267	0.1088	0.07	Cosine Similarity
SSNCSE_NLP	task_1a_2	0.0652	0.0406	0.1004	0.05	TF IDF
IMS_UNIPD	tfidf_lemma	0.0575	0.0324	0.1068	0.02	TF IDF, lemma words
IMS_UNIPD	tfidf_stem	0.056	0.0341	0.1077	0.03	TF IDF, stemmed words
IMS_UNIPD	bm25_lemma	0.0441	0.0143	0.147	0.03	BM 25, lemma forms
fs_hit_2	fs_hit_2_task1a_02	0.0126	0	0.041	0.02	Lucene, TFIDF
Uottawa_NLP	run1_Glove	0.0123	0	0.0222	0	Glove
fs_hit_2	fs_hit_2_task1a_03	0.0123	0	0.0395	0.02	Lucene, Dirichlet Similarity
Uottawa_NLP	run2_Doc2Vec	0.0029	0	0.0051	0	Doc2Vec

participating teams. The comparative results are in Tables 1 and 2. We briefly describe below the methodologies used by each team in each of their runs. Details can be found in the working notes of the respective submissions.

Task 1a : Precedent Retrieval : We describe the methodologies for the task in brief :

- **UB** [9] : The team was from the University of Botswana. In their first submitted run $UB-1$, they weighed the terms in the query and documents using TF-IDF. In their second run, $UB-2$ they extracted key concepts and used TF-IDF for retrieval. The best performance in terms of MAP, BPREFa and recip_rank for the task was by their third run, $UB-3$, where they use Terrier 4.2 KL divergence model.
- **double_liu_2020** [11] : This team is affiliated to the Heilongjiang Institute of Technology, China. They extract the top 50% of the words based on their IDF scores as the search keywords in their first and third runs. In the second run, they used all the words as search keywords. In terms of MAP, BPREF in Task 1a, their third run (double_liu_2020_3) performed the second best.
- **fs_hu** [10] : This team from the Foshan University, China used language model and Dirichlet Smoothing for retrieval. They performed the best in terms of P@10.

- **TUW_Informatics** [8]: This team from TU Wien experiment with different stopword lists. In the first run, basic, preprocessing is performed on the documents and retrieval is through BM-25 algorithm. In their next two runs, word_count and false_friends, they experiment with different methods for stopword removal as the preprocessing step.
- **SSNCSE_NLP** [3]: This team is from Sri Siva Subramaniya Nadar College of Engineering. They use BM-25 and TF-IDF for the task.
- **fs_hit_1** [15] : This team is from the Foshan University, China. They used BM25 and TF-IDF similarity in their first and second runs. Their second run was the second best performing method in terms of recip_rank and P@10.
- **fs_hit_2** [16] : This team, also from the Foshan University, China, explored different Language models for the task. In their first run, they use the language model assorting algorithm of Indri. In the second run, language model assorting algorithm of Lucene is used. In the third run, they use language Model with Dirichlet Similarity.
- **IMS_UNIPD** [7]: This team is affiliated to the Information Management System (IMS) Group, University of Padua. They used BM-25 on the lemma forms of the words in the query and candidate case documents for the bm25_lemma run; used TF-IDF weighting on the lemma forms in tfidf_lemma run and TF-IDF weighting on the stemmed words in the tfidf_stem run.
- **Lawnics** [2]: This team is from Lawnics Technologies, India. They use BM25 and topic embedding methods for the precedent retrieval task, for their first and second runs respectively.
- **Uottawa_NLP** [1]: This team is from the University of Ottawa. After preprocessing the documents, they used Glove, Doc2Vec and TF-IDF based methods, for the first, second and third runs respectively.

Task 1b : Statute Retrieval : We describe the methodologies for the task in brief :

- **scnu** [8] : This team has its members from the South China Normal University. It was the only team that modelled the the task as a supervised task and performed training using the training dataset provided. Their first run that uses BERT is the best performing method in the statute retrieval task in terms of MAP, BPREF and P@10. They are second best in terms of recip_rank. They have also experimented with different supervised methods in their second and third runs.
- **SSNCSE_NLP** [3]: This team is from Sri Siva Subramaniya Nadar College of Engineering. They use BM-25 and TF-IDF for the task. They get the second best MAP scores for the task.
- **IMS_UNIPD** [7]: This team is affiliated to the Information Management System (IMS) Group, University of Padua. They used BM-25 on the lemma forms of the words in the query and candidate case documents for the bm25_lemma run; used TF-IDF weighting on the lemma forms in tfidf_lemma run and TF-IDF weighting on the stemmed words in the tfidf_stem run. They perform second best in terms of P@10.

[8] Working note not submitted.

Table 2. Results of Task 1b: Statute retrieval for queries. All measures averaged over 10 test queries. Numbers in **bold** and underline indicate the best and the second-best performing methods corresponding to the evaluation metrics. Rows are sorted in decreasing order of MAP (primary measure).

Team Name	Run_ID	MAP	BPREF	recip_rank	P @ 10	Method Used
scnu	scnu_1	**0.3851**	**0.3054**	0.5615	**0.18**	BERT
SSNCSE_NLP	task_1b_2	0.3423	0.136	0.3423	0.07	TF IDF
IMS_UNIPD	tfidf_stem	0.3383	0.279	0.5349	0.17	TF IDF, stemmed words
IMS_UNIPD	tfidf_lemma	0.3159	0.2568	0.5288	0.17	TF IDF, lemma forms
UB	UB-2	0.3134	0.2633	**0.5787**	0.15	Extract key concepts, TF-IDF
UB	UB-1	0.3085	0.2633	0.573	0.14	TF-IDF
SSN_NLP	R1	0.2975	0.2531	0.4769	0.15	BM 25
LAWNICS	LAWNICS_1	0.2962	0.2812	0.4607	0.13	Preprocessing, BM25
LAWNICS	LAWNICS_2	0.2962	0.2812	0.4607	0.13	Law2Vec embeddings
TUW_informatics	basic	0.2619	0.2033	0.4855	0.13	Preprocessing, BM25
TUW_informatics	word_count	0.2574	0.214	0.3946	0.14	Preprocessing, BM 25
Uottawa_NLP	run3_TFIDF	0.2506	0.186	0.3144	0.12	Preprocessing, TFIDF
fs_hu	fs_hu_task1b	0.235	0.198	0.3581	0.08	TF-IDF, Jaccard
TUW_informatics	false_friends	0.2316	0.1855	0.3814	0.1	Preprocessing, BM 25
IMS_UNIPD	bm25_lemma	0.231	0.1586	0.4595	0.15	BM 25 on lemma forms
fs_hit_1	fs_hit_1_task1b_03	0.2139	0.1587	0.3371	0.13	Language Model with Dirichlet smoothing
fs_hit_2	fs_hit_2_task1b_01	0.2003	0.1587	0.3452	0.1	Language Model
fs_hit_2	fs_hit_2_task1b_03	0.1886	0.132	0.279	0.1	Language Model
UB	UB-3	0.1876	0.1502	0.2468	0.09	Terrier 4.2 KL divergence model
fs_hit_2	fs_hit_2_task1b_02	0.1777	0.1247	0.2546	0.12	Language Model
fs_hit_1	fs_hit_1_task1b_01	0.1703	0.0945	0.2196	0.12	Language Model with JM smoothing
fs_hit_1	fs_hit_1_task1b_02	0.1703	0.0945	0.2196	0.12	Language Model with JM smoothing
Uottawa_NLP	run1_Glove	0.1462	0.084	0.345	0.1	preprocessing, Glove
scnu	scnu_3	0.1301	0.048	0.1531	0.12	Chen at.al., Enhanced LSTM for Natural Language Inference, ACL 2017
SSNCSE_NLP	task_1b_1	0.1181	0.069	0.2739	0.07	BM 25
nlpninjas	nlpninjas_st1	0.0917	0.024	0.1204	0.07	n-gram, BM25
Uottawa_NLP	run2_Doc2Vec	0.0441	0.008	0.067	0.02	Doc2Vec
scnu	scnu_2	0.0254	0	0.0203	0	Xiong et.al., End-to-End Neural Ad-hoc Ranking with Kernel Pooling, SIGIR 2017

- **UB** [9] : The team was from the University of Botswana. In their first submitted run they weighed the terms in the query and documents using Tf-IDF. In the second run, they extracted key concepts and used TF-IDF for retrieval. This method acheives the best recip_rank for the task. In their third run they use Terrier 4.2 KL divergence model.
- **SSN_NLP** [12]: The team has its members from the SSN College Of Engineering. They use BM-25 for the statute retrieval task.
- **Lawnics** [2]: This team is from Lawnics Technologies, India. They use BM25 in their first run. They explore Law2Vec embedings in their second run. They achieve second best BPREF scores for the task.
- **TUW_Informatics** [8]: This team from TU Wien experiment with different stopword lists. In the first run, basic, preprocessing is performed on the documents and retrieval is through BM-25 algorithm. In their next two runs, word_count and false_friends, they experiment with different methods for stopword removal as the preprocessing step.
- **Uottawa_NLP** [1]: This team is from the University of Ottawa. After preprocessing the documents, they used Glove, Doc2Vec and TF-IDF based methods, for the first, second and third runs respectively.
- **fs_hu** [10] : This team from the Foshan University, China used TF-IDF and Jaccard to compute similarity.

- **fs_hit_1** [15] : This team is from the Foshan University, China. They experiment with different Language Models with Dirichlet smoothing and JM Smoothing with different hyper-parameters.
- **fs_hit_2** [16] : This team, also from the Foshan University, China, explored different Language models for the task tuned with different hyper-parameters.
- **nlpninjas** [9] : This team is from Deloitte USI. They combined unigrams and bigrams. They used BM25 for retrieval.

5 Methodologies for Task 2: Rhetorical Role Labeling for Legal Judgements

Table 3. Results of Task 2: Rhetorical Role Labeling for Legal Judgements. Measures averaged over 10 test documents comprising of $1,905$ sentences. Rows are sorted in decreasing order of FScore (primary measure).

Run	Macro Precision	Macro Recall	Macro F-Score	Accuracy	Method
ju_nlp_2	**0.506**	**0.501**	**0.468**	0.588	ROBERTA
ju_nlp_3	0.519	0.479	0.457	0.57	BERT+Logistic Regression
heu_gjm_1	0.541	0.472	0.457	0.603	BERT
ju_nlp_1	0.504	0.483	0.452	0.588	
heu_gjm_2	0.526	0.468	0.451	0.598	
double_liu_3	0.472	0.486	0.444	**0.619**	
heu_gjm_3	0.529	0.456	0.444	0.59	
spectre_1	0.485	0.483	0.442	0.584	
lawnics_2	0.479	0.479	0.435	0.584	
fs_hu_1	0.493	0.454	0.428	0.562	
fs_hit1_3	0.484	0.449	0.41	0.574	
fs_hit2_1	0.411	0.465	0.405	0.535	
fs_hit1_2	0.456	0.433	0.405	0.578	
fs_hit2_2	0.455	0.427	0.398	0.549	
benchmark	0.467	0.418	0.388	0.482	
fs_hit1_1	0.486	0.406	0.385	0.508	
double_liu_2	0.423	0.407	0.355	0.488	
ssncse_nlp_2	0.384	0.4	0.354	0.46	
ssncse_nlp_1	0.473	0.354	0.333	0.467	
double_liu_1	0.432	0.351	0.327	0.469	
fs_hu_2	0.262	0.343	0.266	0.457	
lawnics_1	0.208	0.164	0.119	0.152	

6 Concluding Discussions

The FIRE 2020 AILA track has created benchmark datasets for two important tasks in the field of legal data analytics. We retained AILA 2019's task of retrieving relevant statutes and precedents for a query. We created a new task on rhetorical role labelling of sentences in Indian legal documents. For the precedent

[9] Working note not submitted

retrieval task, we conclude from the results that it is a challenging task, mainly because of the difference in length of the query and a prior-case document. For the statute retrieval task, training BERT using very little amount of data (50 queries and their corresponding gold standard statutes), shows promising results. For the rhetorical role labelling task, participants have used state-of-the-art deep learning techniques like ROBERTA and BERT, which gives good results.

In the future, we plan to extend the dataset of both the tasks.

Acknowledgements: The track organizers thank all the participants for their interest in this track. We also thank the FIRE 2019 organizers for their support in organizing the track.

References

1. Almuslim, I., Inkpen, D.: Document level embeddings for identifying similar legal cases and laws (aila 2020 shared task). In: Proceedings of FIRE 2020 - Forum for Information Retrieval Evaluation (December 2020)
2. Arora, J., Patankar, T., Shah, A., Joshi, S.: Artificial intelligence as legal research assistant. In: Proceedings of FIRE 2020 - Forum for Information Retrieval Evaluation (December 2020)
3. Balaji, N.N.A., Bharathi, B., Bhuvana, J.: Legal information retrieval and rhetorical role labelling for legal judgements. In: Proceedings of FIRE 2020 - Forum for Information Retrieval Evaluation (December 2020)
4. Bhattacharya, P., Ghosh, K., Ghosh, S., Pal, A., Mehta, P., Bhattacharya, A., Majumder, P.: Overview of the FIRE 2019 AILA track: Artificial Intelligence for Legal Assistance. In: Proceedings of FIRE 2019 - Forum for Information Retrieval Evaluation (December 2019)
5. Bhattacharya, P., Hiware, K., Rajgaria, S., Pochhi, N., Ghosh, K., Ghosh, S.: A Comparative Study of Summarization Algorithms Applied to Legal Case Judgments. In: Proc. European Conference on Information Retrieval (ECIR) (2019)
6. Bhattacharya, P., Paul, S., Ghosh, K., Ghosh, S., Wyner, A.: Identification of rhetorical roles of sentences in indian legal judgments. In: Proc. International Conference on Legal Knowledge and Information Systems (JURIX) (2019)
7. Di Nunzio, G.M.: A study on lemma vs stem for legal information retrieval using r tidyverse. In: Proceedings of FIRE 2020 - Forum for Information Retrieval Evaluation (December 2020)
8. Fink, T., Recski, G., Hanbury, A.: Fire2020 aila track: Legal domain search with minimal domain knowledge. In: Proceedings of FIRE 2020 - Forum for Information Retrieval Evaluation (December 2020)
9. Leburu-Dingalo, T., Motlogelwa, N.P., Thuma, E., Modungo, M.: Ub at fire 2020 precedent and statute retrieval. In: Proceedings of FIRE 2020 - Forum for Information Retrieval Evaluation (December 2020)
10. Li, Z., Kong, L.: Language model-based approaches for legal assistance. In: Proceedings of FIRE 2020 - Forum for Information Retrieval Evaluation (December 2020)
11. Liu, L., Liu, L., Han, Z.: Query revaluation method for legal information retrieval. In: Proceedings of FIRE 2020 - Forum for Information Retrieval Evaluation (December 2020)

12. S., K., D., T., Aravindan, C.: Best matching algorithm to identify and rank the relevant statutes. In: Proceedings of FIRE 2020 - Forum for Information Retrieval Evaluation (December 2020)
13. Wang, P., Fan, Y., Niu, S., Yang, Z., Zhang, Y., Guo, J.: Hierarchical matching network for crime classification. In: Proceedings of ACM SIGIR Conference on Research and Development in Information Retrieval. pp. 325–334. SIGIR'19 (2019)
14. Wang, P., Yang, Z., Niu, S., Zhang, Y., Zhang, L., Niu, S.: Modeling dynamic pairwise attention for crime classification over legal articles. In: Proceedings of ACM SIGIR Conference on Research & Development in Information Retrieval. pp. 485–494. SIGIR '18 (2018)
15. Wu, M., Wu, Z., Xiangyu, W., Han, Z.: Retrieval model and classification model for aila2020. In: Proceedings of FIRE 2020 - Forum for Information Retrieval Evaluation (December 2020)
16. Xu, Y., Li, T., Han, Z.: The language model for legal retrieval and bert-based model for rhetorical role labeling for legal judgments. In: Proceedings of FIRE 2020 - Forum for Information Retrieval Evaluation (December 2020)