COLIEE-2018: Evaluation of the Competition on Legal Information Extraction and Entailment

Yoshinobu Kano[6(✉)], Mi-Young Kim[1,2], Masaharu Yoshioka[4,5], Yao Lu[2], Juliano Rabelo[2], Naoki Kiyota[6], Randy Goebel[2,3], and Ken Satoh[7]

[1] Department of Science, Augustana Faculty, University of Alberta, Camrose, AB, Canada
miyoung2@ualberta.ca
[2] Alberta Machine Intelligence Institute, University of Alberta, Edmonton, AB, Canada
{yaol, rabelo, rgoebel}@ualberta.ca
[3] Department of Computing Science, University of Alberta, Edmonton, AB, Canada
[4] Graduate School of Information Science and Technology, Hokkaido University, N14 W9, Kita-ku, Sapporo-shi, Hokkaido, Japan
yoshioka@ist.hokudai.ac.jp
[5] Global Station for Big Date and Cybersecurity, Global Institution for Collaborative, Research and Education, Hokkaido University, Kita-ku, Sapporo-shi, Hokkaido, Japan
[6] Faculty of Informatics, Shizuoka University, 3-5-1 Johoku, Naka-ku, Hamamatsu-shi, Shizuoka, Japan
kano@inf.shizuoka.ac.jp, nkiyota@kanolab.net
[7] National Institute of Informatics, 2-1-2 Hitotsubashi, Chiyoda-ku, Tokyo, Japan
ksatoh@nii.ac.jp

Abstract. We summarize the evaluation of the 5th Competition on Legal Information Extraction/Entailment 2018 (COLIEE-2018). The COLIEE-2018 tasks include two tasks in each of statute law and case law. The case law component includes an information retrieval (Task 1), and the confirmation of an entailment relation between an existing case and an unseen case (Task 2). The statute law component includes information retrieval (Task 3) and entailment/question answering (Task 4). Participation was open to any group based on any approach. 13 teams participated in the case law competition, and we received results from 7 teams where 6 submissions to Task 1 (12 runs), and 4 submissions to Task 2 (8 runs). Regarding the statute law, there were submissions of 17 runs from 8 teams (including 2 organizers' runs) for Task 3 and 7 runs from 3 teams for Task 4. We describe each team's approaches, our official evaluation, and analysis on our data and submission results. We also discuss possibilities for future competition tasks.

Keywords: COLIEE · Legal information retrieval · Legal textual entailment · Legal question answering · AI and law · Juris-informatics

K. Kojima et al. (Eds.): JSAI-isAI 2018 Workshops, LNAI 11717, pp. 177–192, 2019.
https://doi.org/10.1007/978-3-030-31605-1_14

1 Introduction

The Juris-informatics workshop series was created to promote community discussion on both fundamental and practical issues on legal information processing, with the intention to embrace various disciplines, including law, social sciences, information processing, logic and philosophy, including the existing conventional "AI and law" area.

Competition on Legal Information Extraction/Entailment (COLIEE) is a series of evaluation campaigns to discuss the state of the art for information retrieval and entailment using legal texts [1–3]. In the previous COLIEE 2014–2017, there were two tasks (information retrieval (IR) and entailment) using Japanese Statue Law (civil law). In COLIEE 2018, we conduct new two tasks (IR and entailment) for using Canadian case law (Task 1/2) and two tasks for using Japanese Statue Law that are same settings for the previous campaigns (Task 3/4).

Task 1 is a legal case retrieval task, and it involves reading a new case Q, and extracting supporting cases S1, S2,…, Sn from the provided case law corpus, hypothesized to support the decision for Q. Task 2 is the legal case entailment task, which involves the identification of a paragraph or paragraphs from existing cases, which entail the decision of a new case. For the information retrieval task (Task 3), based on the discussion about the analysis of previous COLIEE IR tasks [4], we modify the evaluation measure of the final results and also ask the participants to submit ranked relevant articles results to discuss the detailed difficulty of the questions. For the entailment task (Task 4), we performed categorized analyses to show different issues of the problems and characteristics of the submissions, in addition to the accuracy evaluation as same as the previous COLIEE tasks.

In the following sections, we will describe each task in detail, explain participants' systems, and assessment results.

2 COLIEE Case Law Competition Tasks

COLIEE-2018 Case Law data is drawn from an existing collection of predominantly Federal Court of Canada case law, provided by vLex Canada (http://ca.vlex.com).

2.1 Task 1: Case Law Retrieval Task

Our goal is to explore and evaluate case law retrieval technologies that are both effective and reliable. The task investigates the performance of systems that search a set of legal cases that support a previously unseen case description. The goal of the task is to accept a query and return noticed cases in the given collection. We say a case is 'noticed' with respect to a query *iff* the case supports the decision of the query case. In this task, the query case does not include a decision, because our goal is to determine how accurately a machine can capture decision-supporting cases for a new case (with no decision).

The process of executing the new query cases over the existing cases and then generating the experimental runs should be entirely automatic. In the training data, each query case is used with a pool of legal cases, and the noticed cases in the pool are produced as the answer. In test data, only query cases and a pool of case laws will be included, with no noticed case information.

The format of the COLIEE case law competition data in Task 1 is as follows:

<**pair id="t1-1">**
<**query content_type="summary" description="The summary of the case created by human expert.**">
The parties to this consolidated litigation over the drug at issue brought reciprocal motions, seeking that the opposing party be compelled to provide a further and better affidavit of documents ... (omitted)
</**query**>
<**query content_type="fact" description="The facts of the case created by human expert.**">
[1] Tabib, Prothonotary: The Rules relating to affidavits of documents should be well known by litigants. Yet it seems that parties are either not following them strictly, or are assuming that others are not ... (omitted)
</query>
<**cases_noticed description="The corresponding case id in the candidate cases**">
18,45,130
</cases_noticed>
<**candidiate_cases description="The candidate cases indexed by id**">
<**candidate_case id="0**"> Case cited by: 2 cases Charest v. Can. (1993)....(omitted)
</candidate_case>
<**candidate_case id="1">** Case cited by: one case Chehade, Re (1994), 83 F.T.R. 154 (TD) ... (omitted)
</candidate_case>
... (omitted)
<**candidate_case id="199">** Desjardins v. Can. (A.G.) (2004), 260 F.T.R. 248 (FC) MLB headnote ... (omitted)
</candidate_case>
</candidate_cases> </pair>

The above is an example of Task 1 training data where query id "t1-1" has 3 noticed cases (IDs: 18, 45, 130) out of 200 candidate cases. The test corpora will not include a <cases_noticed> tag information. Out of the given candidate cases for each query, participants are required to retrieve noticed cases.

2.2 Task 2: Case Law Entailment Task

Our goal in Task 2 is to predict the decision of a new case by entailment from previous relevant cases. As a simpler version of predicting a decision, a decision of a new case and a noticed case will be given as a query. Then a case law textual entailment system must identify which paragraph in the noticed case entails the decision, by comparing the extracting and comparing the meanings of the query and paragraph.

The task evaluation measures the performance of systems that identify a paragraph that entails the decision of an unseen case. Training data consists of a triple: a query, a noticed case, and a paragraph number of the noticed case by which the decision of the query is allegedly entailed. The process of executing queries over the noticed cases and generating the experimental runs should be entirely automatic. Test data will include only queries and noticed cases, but no paragraph numbers.

The format of the COLIEE competition data in Task 2 is as following:

<**pair id="t2-1**">
<**query**>
<**case_description content_type="summary" description="The summary of the case created by human expert."**>
The applicant owned and operated the Inn on the Park Hotel and the Holiday Inn in Toronto ... (omitted)
</case_description>
<**case_description content_type="fact" description="The facts of the case created by human expert."**>
... </**case_description**>
<**decision description="The decision of the query case."**> The applicant submits that it is unreasonable to require the applicant to produce the information and documentation referred to in the domestic Requirement Letter within 62 days ... (omitted)
</decision>
<**cases_noticed description="The supporting case of the basic case**">
<**paragraph paragraph_id="1"**>
[1] Carruthers, C.J.P.E.I. : This appeal concerns the right of the Minister of National Revenue to request information from an individual pursuant to the provisions of s. 231.2(1) of the Income Tax Act , S.C. 1970-71-72, c. 63. Background
</paragraph>
<**paragraph paragraph_id="2"**>
[2] The appellant, Hubert Pierlot, is the main officer and shareholder of Pierlot Family Farm Ltd. which carries on a farm operation in Green Meadows, Prince Edward Island.
</paragraph>
... (omitted)
<**paragraph paragraph_id="26"**>
[26] I would, therefore, dismiss the appeal. Appeal dismissed. Editor: Steven C. McMinniman/vem [End of document]
</**paragraph**>
</cases_noticed>
</query>
<**entailing_paragraph description="The paragraph id of the entailed case."**>**13**</**entailing_paragraph**>
</pair>

The above is an example of Task 2 training data, and the example says that a decision in the query was entailed from the paragraph No. 13 in the given noticed case. The decision in the query does not comprise the whole decision of the case. This is a decision for a portion of the case, and a paragraph that supports the decision should be

Table 1. Baseline performances of Tasks 1 and 2

Tasks	Task 1	Task 2
Precision of term cosine similarity	0.2649	0.0405
Recall of term cosine similarity	0.4102	0.5094
F-measure of term cosine similarity	0.3219	0.0751

identified in the given noticed case. The test corpora will not include the <entailing_paragraph> tag information, and participants are required to identify the paragraph number which entails the query decision.

2.3 Evaluation Metrics and Baselines

The measures for ranking competition participants are intended only to calibrate the set of competition submissions, rather than provide any deep performance measure. The data sets for Tasks 1 and 2 are annotated, so simple information retrieval measures (precision, recall, F1-measure, accuracy) can be used to rank each submission. Task 1 calculates these measures based on number of cases for all queries, while Task 2 based on number of paragraphs for all queries. For Tasks 1 and 2, we consider the term cosine similarity as the baseline model. Table 1 presents the performances of the baseline model.

2.4 Submitted Runs and Results

In the overall case law competition, 13 teams registered, 6 teams submitted their system results in Task 1 (for a total of 12 runs), and 4 teams submitted their results in Task 2 (for a total of 8 runs). Some participants submitted multiple runs for a task. We present the results achieved by runs against the Information Retrieval and Entailment subtasks in Tables 2 and 3, respectively.

Draijer and Verberne (system id: UL) [5] used Random Forest with eight different features for Task 1. The eight features are More Like This Score on Facts, More Like This Score on Summary, Doc2vec Cosine Similarity distance to Facts, Doc2vec Cosine Similarity distance to Summary, TF-IDF Euclidean distance to Facts, TF-IDF Euclidean distance to Summary, TF-IDF Cosine similarity distance to Facts, and TF-IDF Cosine similarity distance to Summary.

Chen et al. (system id: Smartlaw) [6] proposed using association rules in both Tasks 1 and 2. They first experimented with a machine learning-based model adopting Word2Vec/Doc2Vec as features. But machine learning methods have several disadvantages for this task: first, the tasks have very limited training samples, which make current machine learning models hard to achieve good performance. Second, the space consumption of datasets and the computational cost of training exponentially increase when the size of data expands. To enhance the scalability of the solutions, they propose two association rule models: what is labelled as basic association rule model, and another co-occurrence association rule model. The basic association rule model considers only the similarity between the source document and the target document, and it does not leverage a manually labeled relevancy dictionary. The co-occurrence association rule model uses a relevancy dictionary in addition to the basic association rule model.

Tran et al. (system id: JNLP) [7] explored benefits from analyzing legal documents' summaries and logical structures for Task 1. They extended the summary of both the query and the candidates to include more attributes from fact/paragraphs. They propose to obtain document embedding information guided by the document summary. This information is used to estimate the phrasal scores for each document given their summary and paragraphs. Subsequently, they train the model with the summary acting

as gold catchphrases and paragraphs acting as document sentences. After building the trained model, they generate a latent summary in continuous vector space. For the ranking of candidates, they use two selection strategies: hard top k, and flexible bound relative to score deviation.

UNCC0 applied ensemble learning using the following classifiers: logistic regression, XGBoost, Random forest, and Support Vector Machine classifier. They used resampling of input data using jnlp SMOTE for further training.

Yoshioka and Song (system id: HUKB) [8] built an IR system for the Task 1 by using the following two steps to retrieve the referred cases: first (1) they build a ranked retrieval, using an IR system to rank candidates. Since the input queries are full text case laws consisting of several parts (summary, citations, paragraph list, etc.), they experimented using different parts for building the target database and the queries. They also analyzed the effect of building one database per query (using only the given candidates for that query), and then building one database using all candidates. Their best performance was achieved when the database used all available case parts; the queries used only the summary and the database was constructed with all candidates. In their second technique (2) from a selection of the referred cases, they choose which of those cases returned in step (1) are going to be used as their system's answer. They tried two strategies: first, select the top n ranked cases (n fixed a priori), then select a variable number of cases by checking the similarity with non-related cases.

Rabelo et al. (system id: UA) [9] modeled Tasks 1 and 2 as binary classification problems. For Task 1, they constructed feature matrices by using a cosine similarity measure between paragraphs from the query case and each candidate case. Those matrices were then transformed into fixed size feature vectors via a histogram approach with pre-determined score bounds, and given to a Random Forest classifier. They also applied post processing to leverage statistical a priori knowledge. Since the dataset in Task 1 is very imbalanced, they under-sampled the dominant class and over-sampled the rarer class by synthesising samples with SMOTE. Their approach for Task 2 was also based on extracting similarity-based features from the query and noticed cases, and feeding those features to a Random Forest classifier.

Lefoane et al. (system id: UBIRLED) [10] propose an approach based on Information Retrieval and unsupervised learning to Task 1: TFIDF is used as a similarity measure between a query and candidate cases. A k-nearest neighbor search with TFIDF as a distance measure is also used. They first rank documents according to their relevance to the query, then apply filtering to exclude the lowest scoring documents from relevant cases, using a threshold value to cut off non-relevant case judgments.

In Table 2, we can see that most systems show better performance than the baseline model. The JNLP system shows the best performance combining lexical features and latent features embedding summary properties (limiting the average number of noticed cases to 10), and it achieved significant increase of the F-measure compared to other systems.

HUKB1 and HUKB2 systems extracted 194 and 191 cases as noticed cases. JNLP-r = 2.5 and JNLP-k = 10 systems extracted 412 and 399 cases. The Smartlaw system extracted 271 cases, UA, UA-postproc, and UA-smote systems extracted 203, 254, and 247 cases, UBIRLED-1, UBIRLED-2, and UBIRLED-3 systems extracted 392, 453, and 64 cases, and UL system extracted 190 cases. Even though JNLP systems extracted the

most cases amongst the systems, they showed the best precision performance. In Task 1, many participants used machine learning classifiers, but the system which used more sophisticated features such as a combination of lexical features and latent features embedding summary properties showed the best performance in this year's competition.

Table 3 reports the results of Task 2, where UA and UA-500 showed the best performance, which is significantly better than the baseline performance. The UA and UA-500 systems used similarity-based features input to a Random Forest classifier with different number of estimators. Among the 8 systems, 6 systems showed better performance than the baseline model on Task 2. Task 2 was much difficult than Task 1, and even humans have difficulty in choosing the correct paragraph with the appropriate entailment relations. We can also see the task is difficult based on the low performance on all the systems.

The Tasks 1 and 2 have been newly created in this year's competition, and we think there are many rooms for improvement, such as the evaluation method of Task 2, imbalanced data set, small size set of data which have limitations in applying machine learning techniques, etc. We hope to solve these limitations step-by-step for next competition, to get more robust performances for each task.

Table 2. IR results (Task 1) on the formal run data

Run	Prec.	Recall	F-m.	Run	Prec.	Recall	F-m.
Baseline	0.2649	0.4102	0.3219	UA-postproc	0.3484	0.4038	0.3741
HUKB1	0.4974	0.3084	0.3808	UA-smote	0.3539	0.3927	0.3723
HUKB2	0.4047	0.3037	0.3470	UBIRLED-1	0.1329	0.6232	0.2191
JNLP-r = 2.5	0.5464	0.6550	0.5958	UBIRLED-2	0.1955	0.7202	0.3075
JNLP-k = 10	**0.6763**	**0.6343**	**0.6546**	UBIRLED-3	0.5614	0.1017	0.1723
Smartlaw	0.2871	0.4308	0.3446	UL	0.5638	0.3021	0.3934
UA	0.3725	0.3227	0.3458				

Table 3. Entailment results (Task 2) on the formal run data

Run	Prec.	Recall	F-m.	Run	Prec.	Recall	F-m.
Baseline	0.0405	0.5094	0.0751	UBIRLED-1	0.0484	0.8302	0.0914
Smartlaw	0.0465	0.1509	0.0711	UBIRLED-1	0.0495	0.9245	0.0940
UA	**0.2381**	**0.2830**	**0.2586**	UBIRLED-1	0.0467	0.7925	0.0881
UA-100	0.1905	0.2264	0.2069	UNCC0	0.0330	0.0566	0.0417
UA-500	**0.2381**	**0.2830**	**0.2586**				

3 COLIEE Statute Law Competition Tasks

For the statute law tasks, training and test data of the legal questions are collected from the civil law short answer (multiple choice) part of the Japanese legal bar exam. All questions and Japanese civil law articles (total 1056 articles) are provided in two

languages; Japanese and English. English version of the Law articles and questions are provided by the organizers. The organizers provides data set used for previous campaigns [1–3] as training data (651 questions) and new questions selected from bar exam on 2017 as test data (69 questions both for Task 3 and Task 4 individually).

3.1 Task 3: Statute Law Information Retrieval Task

Task 3 is a task to retrieve articles to decide the appropriateness of the legal question. The participants are asked to submit relevant articles for the questions using Japanese or English data. Each participant can submit at most 3 runs for Task 3. Since most of the system returns only 1 article for each question, the numbers of relevant article(s) for the question affect the system performance. Followings are numbers of questions classified by the number of relevant article.

3.1.1 Submitted Runs

Following 8 teams (alphabetical order except organizers' team for baseline) submitted the results. Since all team can submit at most three runs, there are 17 runs in total. Three teams (HUKB, JNLP, and UA) have an experience on submitting results in previous campaign and four teams (Smartlaw, SPABS, UB and UE) are new to the campaign.

HUKB (2 runs) [8] use structural analysis results (condition, decision) of the article and questions and use Indri [11] to calculate similarity measure among different parts. SVM-rank [12] is used to aggregate such similarity measure. HUKB1 decides the number of returned articles based on the analysis of IR retrieval difficulty. HUKB2 returns only 1 article for each question.

JNLP (2 runs) [7] uses structural analysis results (requisite and effectuation) of articles, uses TF-IDF based vector space model for calculating similarity among them. JNLP1 uses similarity between query and articles only for article ranking. JNLP2 calculate final similarity value as a linear combination of similarity used for JNLP1 and similarity between query and article effectuation part. Both runs returns two articles for all questions based on the analysis of training data.

Smartlaw (3 runs) [6] calculate the similarity of a question and an article by checking the similarity between (1–4) gram sets extracted from the question and the article. Based on the experimental analysis, they submit three runs whose setting for constructing (1–4) gram sets are different; Smartlaw, Smartlaw 2 gram, and Smartlaw 3 gram use bigram+trigram, bigram and trigram, respectively.

SPABS (3 runs) uses recurrent neural network to calculate similarity between question and articles. For training word embedding they use English legal documents with Word2Vec. SPABS bm25 is their baseline results using BM25.

UA (1 run) [13] uses same system for COLIEE 2017 for Task 3. This system uses TF-IDF model of Lucene (https://lucene.apache.org/).

UB (3 runs) uses Terrier 4.2 (http://terrier.org/) with PL2 term weighting model as IR platform. UB3 use TagCrowd (https://tagcrowd.com/) to select important keywords from each question and use them as a query of the IR platform. UB2 uses query expansion after UB3 retrieval, and UB1 uses word embeddings.

UE (1 run) uses rule based method to retrieve relevant documents.

ORG (2 runs) uses Indri [11] with simple setting (use question as query and each articles with title are indexed as a document) [7].

Teams who participated previous COLIEE propose an extension or equivalent system for Task 3, and new teams propose methods that are different from previous ones.

3.1.2 Evaluation of Submitted Runs

Table 4 shows the evaluation results of submitted runs including organizer runs. Official evaluation measures are F2 measure, precision (Prec.), recall (Rec.). "ret.", and "rel." represent number of return articles and number of returned relevant articles, respectively. Columns after MAP will be explained later. There are two differences on evaluation measure used in the task compared to the former campaigns:

1. F2 measure, $F2 = (5 \times Prec \times Rec)/(4 \times Prec + Rec)$, is used instead of F1 measure. F2 measure is a variation of f-measure that weights recall higher than precision. If we assume IR task is a preprocess to provide relevant article(s) to the entailment system, it is requested to provide a set of candidate article(s) including relevant article(s) to the entailment system.
2. Macro average is used instead of micro average (Average of evaluation measures are calculated based on the aggregated numbers of relevant articles, returned articles, and returned relevant articles for all questions) used in the former campaigns. Micro average is not so appropriate for the case with different numbers of relevant articles. For example, for analyzing the recall, questions with multiple relevant articles is more important than one with one relevant article. In addition, when the system returns many articles for one query due to the uncertainty of the returned results, this seriously deteriorates the precision of micro average. However, using macro average (Each evaluation measure is calculated based on the numbers of relevant articles, returned articles, and returned relevant articles for each question. After calculating evaluation measure for each question, average of such measure over all questions are calculated), we can reduce the effect of such different characteristics among all retrieved results.

In the previous campaigns, since most of the teams submit only one or two articles for each question, we can only evaluate the topic difficulties based on the number of systems that can return such articles as relevant one. However, it is almost impossible to estimate the reason of the problem. For example, some questions have difficulties to rank the relevant articles higher due to the vocabulary mismatch, and some questions have difficulties to select appropriate one from similar articles (relevant articles are ranked higher but not 1st rank). Therefore, we decide to ask participants to submit long ranking list (100 articles) in addition to the selected relevant article candidate list.

This list provides information that can discuss the type of difficulties to retrieve relevant articles. For the long list, mean average precision (MAP), recall at using top k rank documents as returned documents (R_k) are used for the evaluation measure.

Table 4 also shows information about the evaluation measure for long rank list. However, UE does not submit this long list, values are described as "-".

Table 4. Evaluation of submitted runs (Task3) and organization run

Run id	Language	Ret.	Rel.	F2	Prec.	Rec.	MAP	R_5	R_{10}	R_{30}
UB3	E	69	54	**0.6964**	**0.7826**	0.6860	**0.7988**	0.7978	0.8539	**0.9551**
UA	E	69	50	0.6602	0.7246	0.6522	0.7451	0.7303	0.7528	0.8539
ORGE1	E	69	49	0.6368	0.7101	0.628	0.7381	0.7528	0.809	0.8989
UB2	E	69	47	0.6232	0.6812	0.6159	0.7542	**0.7978**	**0.8652**	**0.9551**
JNLP1	E	138	57	0.6118	0.413	**0.7126**	0.7398	0.764	0.8202	0.9213
Smartlaw	E	138	57	0.6042	0.413	0.7005	0.7036	0.7079	0.764	0.8315
JNLP2	E	138	56	0.5997	0.4058	0.6981	0.7296	0.7528	0.809	0.9101
SPABS_bm25	E	138	55	0.5821	0.3986	0.6739	0.707	0.7753	0.8202	0.9101
UE	E	69	34	0.4516	0.4928	0.4469	–	–	–	–
Smartlaw_3 gram	E	69	34	0.4387	0.4928	0.4324	0.47	0.4494	0.4607	0.5056
UB1	E	69	31	0.4171	0.4493	0.413	0.5355	0.573	0.7191	0.8202
Smartlaw_2 gram	E	141	34	0.3421	0.3023	0.4275	0.4594	0.4382	0.4831	0.5169
SPABS_rnnen	E	138	19	0.215	0.1377	0.2536	0.2638	0.3371	0.4494	0.573
SPABS_rnnsq	E	138	17	0.1957	0.1232	0.2319	0.2662	0.3483	0.4494	0.6067
HUKB2	J	69	53	0.6859	0.7681	0.6763	0.7805	0.7865	0.8427	0.9326
HUKB1	J	74	53	0.6826	0.7536	0.6763	0.7805	0.7865	0.8427	0.9326
ORGJ1	J	69	51	0.6633	0.7391	0.6546	0.7703	0.7753	0.8427	0.9326

Based on the comparison of ORGJ1 and ORGE1, we confirm there is not so big difference between English and Japanese data.

Since average of the relevant articles per query is 1.29 (89/69), the performance of systems that return 2 articles for each question are worse than one that return 1 article only. The best performance system is UB3 that uses tag cloud algorithm to select appropriate keywords for constructing query and use Terrier IR platform to retrieve final results. Teams that have participated in the previous campaigns have almost similar scores except JNLP that returns 2 articles for each question. The performances of new teams except UB are worse than baseline system.

We discuss the difficulties of the questions based on the averaged evaluation measure among team top run results for each language (8 results; HUKB2, JNLP1, SPABS bm25, UB3, UA, Smartlaw, ORGJ, and ORGE). For the questions that have 1 relevant article, 28 out of 51 questions have average MAP = 1.0. It means those questions are easy questions and none of the system made mistake to rank relevant articles as 1st article. For those questions, the system that returns two articles for each question takes bad precision score (precision = 0.5) even though the systems rank the relevant article as 1st rank article. Since those easy questions are not worthwhile to discuss in detail, we only focus on the non-easy questions.

Figure 1 shows averages of MAP, R5, R10 for the non-easy questions (23 questions) with single relevant article. Most of the cases, all of the system find the articles as higher ranked articles (14 questions have R5 = 1 and 2 questions have R5 = 0.875 that means only 1 system cannot rank the articles in top 5). There are few questions that have difficulties to rank relevant articles higher.

Figure 2 shows averages of precision, recall, MAP, R5, R10 for questions with multiple relevant article (2 questions H29-28-E and H29-35-I have three relevant

articles and 16 other questions have two relevant articles). There are few questions where both 1st and 2nd ranked articles are relevant articles (MAP = 1). In other cases, there are many questions whose contents is similar to one of the relevant article, but the other is not so similar.

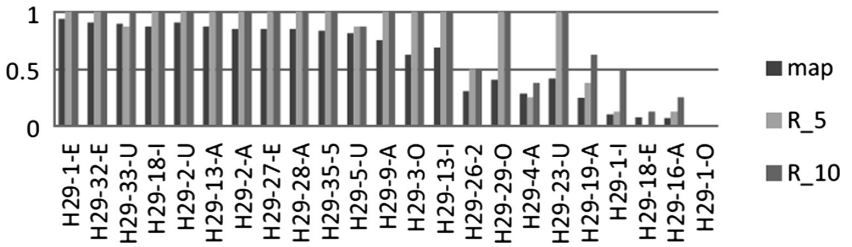

Fig. 1. Averages of MAP, R5, R10 for the non-easy questions with single relevant article

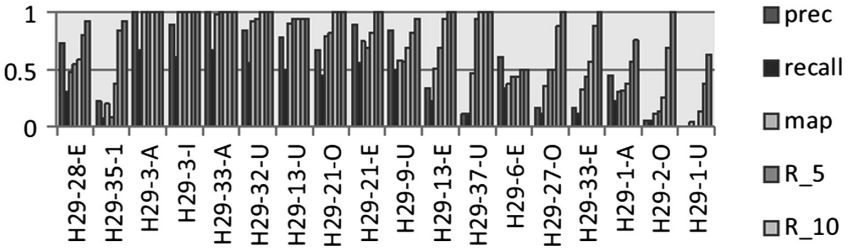

Fig. 2. Averages of precision, recall, MAP, R5, R10 for the non-easy questions with single relevant article

3.1.3 Discussion

Since we have conducted series of campaigns to retrieve relevant articles to entail the questions of Japanese bar exam, most of the system succeed to retrieve relevant articles of the simple questions that have only one relevant article and higher vocabulary (phrase) overlap between question and the relevant article. However, retrieval performance of the questions with vocabulary mismatch is not so good semantic matching technique including RNN approach may be a good approach to tackle this type of problem. But in order to avoid the side effect of degrading the retrieval performance of easy question, preprocessing would be useful to select whether it is necessary to use such semantic matching technique.

For the questions with multiple answers, there are many questions that contents based similarity is not good enough to find out 2nd or 3rd supplemental relevant articles. Information about relationship among articles may be a candidate information resource that are not well utilized at this moment, but further discussion is necessary to tackle this type of the problem.

3.2 Task 4: Statute Law Entailment/Question Answering Task

Task 4 is a task to determine entailment relationships between a given problem sentences and article sentences. Participants should answer yes or no regarding the given problem sentences. There were pure entailment tasks hold until COLIEE 2016, where t1 (relevant article sentences) and t2 (problem sentence) were given. Due to the limited number of available problems, COLIEE 2017 and 2018 did not hold this style of task. In Task 4 of COLIEE 2018, t1 (relevant articles) is not given, participants should find the relevant articles by themselves.

3.2.1 Submitted Runs and Evaluation Results

Following 3 teams submitted the results. Since a team submitted five runs, there are 7 runs in total. Two teams (KIS and UA) have experiences on submitting results in previous tasks and a team (UE) is new to our tasks.

KIS (3 runs) [14] analyze Japanese sentences linguistically, use predicate argument structures to determine similarities. [15] uses frame information to calculate similarity between predicates. Their final results were ensemble of these different modules by SVM.

UA (1 run) [9] uses almost same system of COLIEE 2017 for Task 4. Their system uses condition/conclusion/exception detection rules, and negation dictionaries created manually.

UE (1 run) combined deep neural network with additional features, and word2vec to gain the corresponding civil law articles.

Table 5 shows an evaluation results of submitted runs. Official evaluation measures used in this task is accuracy.

Table 5. Evaluation results of submitted runs (Task 4) and baseline result

Team	Language	Correct Answers (69 questions in total)	Accuracy
BaseLine	N/A	35 (answers No to all)	0.5072
UA	?	44	0.6377
KIS_Frame	Japanese	39	0.5652
KIS_mo3	Japanese	38	0.5507
KIS_dict	Japanese	37	0.5362
KIS_SVM	Japanese	36	0.5217
KIS_Frame2	Japanese	35	0.5072
UE	English	33	0.4783

The best system was UA, which accuracy was 0.6377. The baseline was almost 0.5, because this task is a binary classification, with 35/69 questions are No. Effect of language difference is unclear. In our statue law tasks, the Japanese legal bar exam is the original data, which is translated into English manually. Team UA used translation system and Korean parser internally. Translation process might have absorbed ambiguities and paraphrases.

Because an entailment task is essentially a complex compositions of different subtasks, we manually categorized our test data into categories, depending on what sort of technical issues are required to be resolved. Table 6 shows our categorization results. As this is a compositional task, overlap is allowed between categories. Our categorization is based on the original Japanese version of the legal bar exam.

We have summarized the results of the COLIEE-2018 competition. Two tasks for Case Law, Task 1: retrieving noticed cases (information retrieval), and Task 2: extracting paragraphs of relevant case which entail the conclusion of a new case. Other two tasks for Statute Law, Task 3: information retrieval, and Task 4: entailment/question answering. There were 13 teams who participated in this competition, and we received results from 7 teams. There were 6 submissions to Task 1 (for a total of 12 runs), and 4 submissions to Task 2 (for a total of 8 runs). There are 17 run submissions from 8 teams (including 2 organizers' run) for Task 3 and 7 run submissions from 3 teams for Task 4.

A variety of methods were used for Task 1: combining lexical features and latent features embedding summary properties, creating queries from the summaries of cases, and building an information retrieval system to extract noticed cases, co-occurrence association model, pairwise paragraph similarity computation, K-NN, TF-IDF, and a Random forest classifier. Various features were also proposed: features from summary properties, Word2Vec, Doc2Vec, More Like This Score, cosine similarity, Euclidean distance, etc. For Task 2, co-occurrence association model, similarity-based features fed to a random forest classifier, and ensemble machine learning with SMOTE resembling techniques were used. Even though most systems outperformed baseline, all the performances are low, and the task didn't make it easy to identify relevant useful attributes. For future competitions, we will need to expand the data sets in order to improve the robustness of results. We also need to more deeply investigate how to extract good features for Task 2.

For Task 3, we found there are three types of problem in the test data; i.e., easy question, difficult questions with vocabulary mismatch, and questions with multiple answers. Most of the submission systems are good at retrieving relevant answers for easy questions, but it is still difficult to retrieve relevant articles with other question types. It may be necessary to focus on such question types to improve the overall performance of the IR system. For Task 4, overall performance of the submissions is still not sufficient to use their systems for the real application. However, detailed analysis could capture the characteristics of the submitted systems. We found this task is still a challenging task to discuss and develop deep semantic analysis issues in the real application, and natural language processing in general.

3.2.2 Discussion

Our categorization shown in the previous section suggests several issues and analyses. The largest number among these categories was for the conditions. UA, the best team, was better in this condition category. Their condition detection should have successfully performed. KIS Frame2, which used the frame information, was good in case roles, person relations, and person roles. Their frame relation would have certain effect in these deep semantic issues.

Table 6. Technical category statistics of questions, and correct answers of submitted runs for each category. Team names stand for their number of correct answers for corresponding category.

Category	Team														
	# of questions	UA	Accuracy	UE	Accuracy	KIS_mo3	Accuracy	KIS_dict	Accuracy	KIS_SVM	Accuracy	KIS_Frame2	Accuracy	KIS_Frame	Accuracy
Itemized	3	1	0.33	2	0.67	1	0.33	1	0.33	1	0.33	1	0.33	2	0.67
Numerical priority	3	2	0.67	2	0.67	1	0.33	1	0.33	2	0.67	1	0.33	2	0.67
Entailment	5	2	0.4	2	0.4	1	0.2	1	0.2	4	0.8	2	0.4	2	0.4
Dependency	5	3	0.6	1	0.2	2	0.4	2	0.4	3	0.6	0	0	4	0.8
Article search	5	3	0.6	2	0.4	3	0.6	3	0.6	1	0.2	1	0.2	4	0.8
Paraphrase	5	2	0.4	4	0.8	3	0.6	3	0.6	2	0.4	3	0.6	3	0.6
Negation	7	5	0.71	3	0.43	5	0.71	5	0.71	2	0.29	1	0.14	7	1
Legal terms	7	4	0.57	2	0.29	2	0.29	2	0.29	3	0.43	4	0.57	3	0.43
Normal terms	9	5	0.56	5	0.56	4	0.44	4	0.44	5	0.56	6	0.67	4	0.44
Predicate argument	9	8	0.89	3	0.33	5	0.56	5	0.56	5	0.56	4	0.44	5	0.56
Verb paraphrase	13	7	0.54	6	0.46	7	0.54	7	0.54	7	0.54	7	0.54	4	0.31
Case role	15	8	0.53	6	0.4	9	0.6	9	0.6	6	0.4	11	0.73	6	0.4
Ambiguity	17	9	0.53	7	0.41	8	0.47	8	0.47	8	0.47	10	0.59	9	0.53
Anaphora	20	13	0.65	5	0.25	12	0.6	11	0.55	8	0.4	8	0.4	13	0.65
Morpheme	25	18	0.72	16	0.64	20	0.8	19	0.76	10	0.4	16	0.64	16	0.64
Person relationship	26	14	0.54	11	0.42	13	0.5	13	0.5	13	0.5	18	0.69	10	0.38
Person role	27	16	0.59	12	0.44	14	0.52	14	0.52	14	0.52	18	0.67	13	0.48
Conditions	31	19	0.61	9	0.29	13	0.42	12	0.39	16	0.52	11	0.35	16	0.52

Because the distribution of yes/no answers is quite diverse between submissions, an ensemble could performs better results if we could capture meaningful information for each submission.

4 Conclusion

We have summarized the results of the COLIEE-2018 competition. For the case law, Task 1 retrieves noticed cases (information retrieval), Task 2 extracts paragraphs of relevant case which entail the conclusion of a new case. Task 3 is a task to retrieve articles to decide the appropriateness of the legal question and Task 4 is a task to entail whether the legal question is correct or not. 13 teams participated in the case law competition, and we received results from 7 teams where 6 submissions to Task 1 (for a total of 12 runs), and 4 submissions to Task 2 (for a total of 8 runs). Regarding the statute law, there were 17 run submissions from 8 teams (including 2 organizers' run) for Task 3 and 7 run submissions from 3 teams for Task 4.

A variety of methods were used for Task 1: combining lexical features and latent features embedding summary properties, creating queries from the summaries of cases, and building an information retrieval system to extract noticed cases, co-occurrence association model, pairwise paragraph similarity computation, K-NN, TF-IDF, and a Random forest classifier. Various features were also proposed: features from summary properties, Word2Vec, Doc2Vec, More Like This Score, cosine similarity, Euclidean distance, etc. For Task 2, co-occurrence association model, similarity-based features fed to a random forest classifier, and ensemble machine learning with SMOTE resembling techniques were used. Even though most systems outperformed baseline, all the performances are low, and the task didn't make it easy to identify relevant useful attributes. For future competitions, we will need to expand the data sets in order to improve the robustness of results. We also need to more deeply investigate how to extract good features for Task 2.

For Task 3, we found there are three types of problem in the test data; i.e., easy question, difficult questions with vocabulary mismatch, and questions with multiple answers. Most of the submission systems are good at retrieving relevant answers for easy questions, but it is still difficult to retrieve relevant articles with other question types. It may be necessary to focus on such question types to improve the overall performance of the IR system. For Task 4, overall performance of the submissions is still not sufficient to use their systems for the real application. However, detailed analysis could capture the characteristics of the submitted systems. We found this task is still a challenging task to discuss and develop deep semantic analysis issues in the real application, and natural language processing in general.

Acknowledgements. This research was supported by Alberta Machine Intelligence Institute (AMII), National Institute of Informatics, Shizuoka University and Hokkaido University. Thanks to Colin Lachance from vLex for his constant support in the development of the case law data set, and to support from Ross Intelligence and Intellicon. This work was partially supported by JSPS KAKENHI Grant Number 16H01756, 18H0333808, 17H06103, and JST CREST.

References

1. Kim, M.Y., Goebel, R., Satoh, K.: COLIEE-2015: evaluation of legal question answering. In: Ninth International Workshop on Juris-Informatics (JURISIN 2015) (2015)
2. Kim, M.Y., Goebel, R., Kano, Y., Satoh, K.: COLIEE-2016: evaluation of the competition on legal information extraction/entailment. In: Tenth International Workshop on Juris-Informatics (JURISIN 2016) (2016)
3. Kano, Y., Kim, M.Y., Goebel, R., Satoh, K.: Overview of COLIEE 2017. Epic Ser. Comput. **47**, 1–8 (2017)
4. Yoshioka, M.: Analysis of COLIEE information retrieval task data. In: Arai, S., Kojima, K., Mineshima, K., Bekki, D., Satoh, K., Ohta, Y. (eds.) New Frontiers in Artificial Intelligence, pp. 5–19. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-93794-6_1
5. Draijer, W., Verberne, S.: Case law retrieval with doc2vec and elastic search. In: Twelfth International Workshop on Juris-Informatics (JURISIN 2018), 2018
6. Chen, Y., Zhou, Y., Lu, Z., Sun, H., Yang, W.: Legal information retrieval by association rules. In: JURISIN 2018 (2018)
7. Tran, V., Truong, S.N., Le Nguyen, M.: JNLP group: legal information retrieval with summary and logical structure analysis. In: International Workshop on Juris-Informatics (JURISIN 2018) (2018)
8. Yoshioka, M., Song, Z.: HUKB at COLIEE2018 information retrieval task. In: International Workshop on Juris-Informatics (JURISIN 2018) (2018)
9. Rabelo, J., Kim, M.Y., Babiker, H., Goebel, R., Farruque, N.: Legal information extraction and entailment for statute law and case law. In: JURISIN 2018 (2018)
10. Lefoane, M., Koboyatshwene, T., Narasimhan, L.: KNN clustering approach to legal precedence retrieval. In: Twelfth International Workshop on Juris-Informatics (JURISIN 2018) (2018)
11. Strohman, T., Metzler, D., Turtle, H., Croft, W.B.: Indri: a language-model based search engine for complex queries. In: Proceedings of the International Conference on Intelligent Analysis, pp. 2–6 (2005)
12. Joachims, T.: Optimizing search engines using clickthrough data. In: Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 133–142 (2002)
13. Kim, M.Y., Goebel, R.: Two-step cascaded textual entailment for legal bar exam question answering. In: 16th International Conference on Artificial Intelligence and Law (ICAIL 2017), 2017
14. Hoshino, R., Taniguchi, R., Kiyota, N., Kano, Y.: Question answering system for legal bar examination using predicate argument structure. In: Twelfth International Workshop on Juris-Informatics (JURISIN 2018) (2018)
15. Taniguchi, R., Hoshino, R., Kano, Y.: Legal question answering system using framenet. In: Twelfth International Workshop on Juris-Informatics (JURISIN 2018) (2018)