Retrieval Model and Classification Model for AILA2020

Menghan Wu[b], Zhengyu Wu[b], Xiangyu Wang[b], Zhongyuan Han[a,*]

[a] *Foshan University, Foshan, China*
[b] *Heilongjiang Institute of Technology, Harbin, China*

Abstract

In this paper, we introduce our methods used in the Task1 and Task2 evaluations of AILA2020 (Artificial Intelligence for Legal Assistance). For Task1a, we used BM25 and Cosine Similarity to rank the documents. And the language models with Jelinek-Mercer smoothing method and Dirichlet smoothing method are used for Task1b. For the result we submitted for task2, Logistic Regression model based on TF-IDF feature and BERT is used.

Keywords 1

Artificial Intelligence for Legal Assistance, retrieval model, classification model

1. Introduction

In the age of information, studying previous legal cases is very important for judgments. How to quickly and efficiently find similar precedents, and regulations along with the task of document semantic segmentation is particularly important. Therefore, Artificial Intelligence for Legal Assistance 2020 (AILA), treats these tasks [1] as its theme, aims to develop data sets and models to solve these problems.

Task 1: (Precedent & Statute retrieval) The dataset of Task1A is composed of 3257 judgments delivered by the Supreme Court of India. For each query, we need to calculate the most relevant case documents from these 3257 judgments, and then sort them to get the results. Task1B has determined a set of 197 statues (Sections of Acts) from Indian laws. The content of the inquiry is consistent with Task1A. We need to obtain the most relevant regulations for the inquiry.

Task 2 is to classify each sentence into one of the following 7 semantic segments/rhetorical roles through a given document: 1. Facts 2. Rulings by Lower Courts 3. Argument 4. Statute 5. Precedent 6. Ratio of the decision 7. Ruling by Present Court. It can also be understood as a multi-classification task

2. The proposed approaches

2.1. Methods of Identifying Relevant Prior Cases

Forum for Information Retrieval Evaluation 2020, December 16–20, 2020, Hyderabad, India
EMAIL: menghanwu3@gmail.com (M. Wu); wuzhengyu5@outlook.com (Z. Wu); cn.wangxiangyu@foxmail.com (X. Wang); hanzhongyuan@gmail.com (Z. Han*)
ORCID: 0000-0001-8960-9872 (Z. Han*)

For the task of Identifying Relevant Prior Cases, we submitted three sets of run files.

First, we preprocessed the data, used Lucene to index the queried documents, and removed common punctuation marks and stop words during indexing.

The first submission fs_hit_1_task1a_01 uses Lucene combined with BM25Similarity [2]. BM25 uses IDF (Inverse Document Frequency) to distinguish between common words (relatively unimportant) and important words. At the same time, it believes that the more frequently a word in a document appears, the more relevant the document is to the word.

BM25 has two adjustable parameters: k1 and b. The parameter k1 controls the rising speed of the word frequency result in word frequency saturation. The smaller k1 means the faster the saturation change, and vice versa. The default value is 1.2. The parameter b controls the role played by the field length normalization value. The value range of b is (0,1]. When b=0, normalization is disabled, and when b=1.0, it is completely normalized. The default value is 0.75. And in the experiment, we have used the default value, but the optimal value of k1 and b still depends on the document collection, and finally the first set of results submitted k=1.0, b=1.0.

The second submission fs_hit_1_task1a_02 is used the BM25 model. For the selection of parameters, we adjusted the parameters by referring to the training results of fs_hit_1_task1a_01. Prior to this, the first 50% of the query terms were calculated by TF-IDF ((term frequency-inverse document frequency) is selected as a new query to put into the BM25 retrieval model. The parameters k1=2.0 and b=1.0 were modified.

In addition, we also tried the cosine similarity measure. Get the word frequency vector of the sentence through word segmentation, and calculate the similarity, but the result is unsatisfactory. The formula for cosine similarity is as follows:

$$\text{similarity}(\text{A}, \text{B}) = \frac{A \cdot B}{|A| \times |B|} = \frac{\sum_{i=1}^{n} (A_i \times B_i)}{\sqrt{\sum_{i=1}^{n} (A_i)^2} \times \sqrt{\sum_{i=1}^{n} (B_i)^2}} \tag{1}$$

2.2. Methods of Identifying Relevant Statutes

For the task of Identifying Relevant Statutes, we submitted three run files. The first two sets of results use the Language model with Jelinek-Mercer smoothing method [3], and the last set uses the Language model with Dirichlet smoothing method [3].

The document length of task1b is much shorter than task1a, the average length is close to 200 words. In the statistical word results, there must be a lot of sparseness, it is impossible to include all the keywords that can be used to query them, and the inclusion of certain keywords, the relevance between keyword and document is relatively weak. Therefore, in order to improve the accuracy of the query, language models are used and smoothed.

Before the calculation, we processed the documents as follows: Some common punctuation marks were removed when indexing, and stop words were also removed. In fs_hit_1_task1b_01 and fs_hit_1_task1b_02, we used Jelinek Mercer smoothing with different parameter λ. The value of λ is between (0,1). The choice of λ parameter is very important for the model. The best value depends on the specific collection and query. The closer λ is to 1, the greater the smoothness effect. During our experiment, when $\lambda=1\text{-}10^{-5}$(fs_hit_1_task1b_01) results the best result, we also submitted the results when $\lambda=1\text{-}10^{-6}$(fs_hit_1_task1b_02).

The last set of results submitted uses the default value of Dirichlet smoothing.

2.3. Methods of Rhetorical Role Labeling for Legal Judgements

For the task of Rhetorical Role Labeling for Legal Judgements, the file named fs_hit_1_task2_01 uses the Logistic Regression with the feature of TF-IDF which provided by Scikit-learn.

The rest of two submit, fs_hit_1_task2_02 and fs_hit_1_task2_03, are generated by Bert [4] with different random seeds. In this method, we did not do any processing on the data. When the test set is not released, a random 20% of train set is selected as the test set. On the pre-training model, we chose L-12_H-768_A-12 (BERT-Base). We finetuned it with the train set. When the train epoch is greater than 5, the model is overfitting.

3. Experimental Setting

3.1. Parameter Selection

For Task1A, the k1 and b we used in BM25Similarity are listed as follow.

Table 1

The result of different parameters k1 and b in BM25Similarity

Parameters	MAP	P@10
K1=1.2, b=0.75	0.1082	0.06
K1=1.0, b=0.75	0.1108	0.06
K1=1.0, b=1.0	0.1220	0.06

Table 2

The result of different parameters k1 and b in BM25 and TF-IDF

Parameters	MAP	P@10
K1=1.2, b=0.75	0.1120	0.06
K1=2.0, b=1.0	0.1281	0.07

For Task1B, we selected Jelinek-Mercer smoothing method with different lambda.

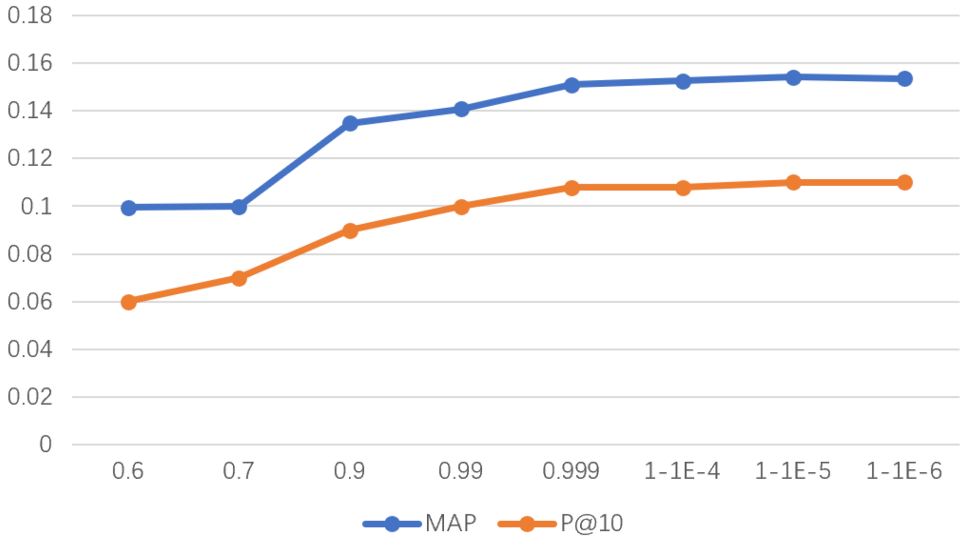

Figure 1

Experimental results with different Lambda in TaskB

After knowing the experimental results, we used some other parameters for Language Model with Dirichlet smoothing. The specific parameters and results are shown in the figure:

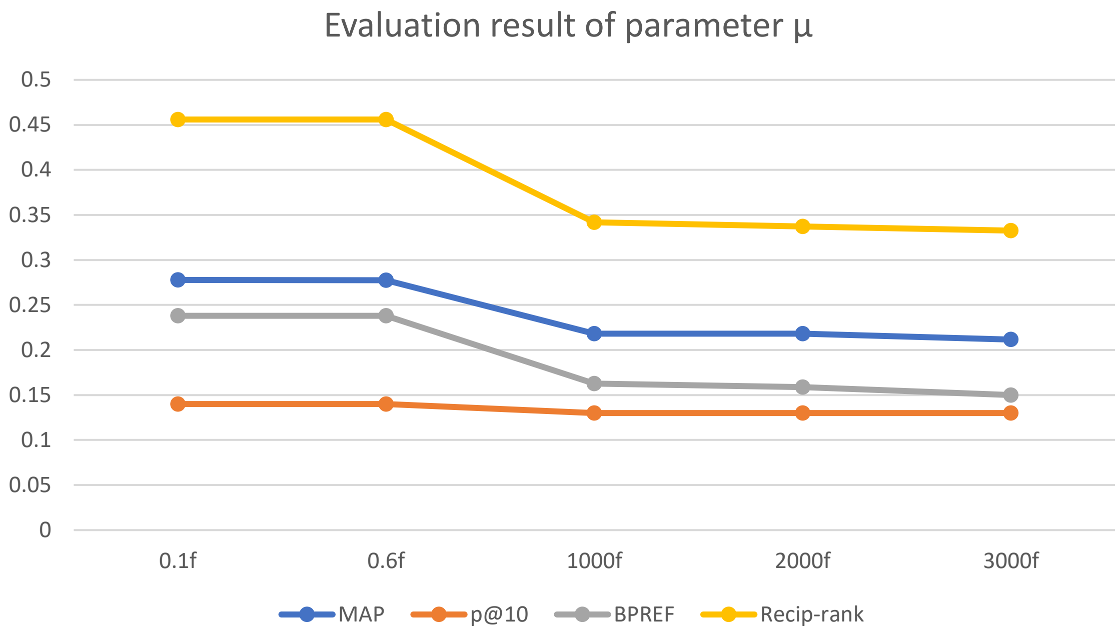

Figure 2
Experimental results with different μ for Language Model with Dirichlet smoothing in TaskB.

For Language Model with Dirichlet smoothing, the default parameter mu of this method is 2000 and it is also the parameter we use to submit the results. We tried experiments when the parameter was close to 0, and the result was slightly higher than the instantiation similarity using the default mu value of 2000.

3.2. Experimental Results

Table 3
Results of the AILA Task 1 - Precedent Retrieval ranked by MAP

Run_ID	MAP	BPREF	recip_rank	P@10
fs_hit_1_task1a_01	0.1294	0.0877	0.1876	0.07
fs_hit_1_task1a_02	0.1215	0.0699	0.2078	0.09
fs_hit_1_task1a_03	0.0696	0.0267	0.1088	0.07

From the experimental results that BM25Similarity has achieved good result, which wins sixth place on MAP and BPREF. When using BM25+TF-IDF the result is lower than first run file there is not much difference in results.

Among the evaluation metrics of the fs_hit_1_task1a_02 file, the metrics p@10 of 0.09 won the second place, and the metrics recip_rank of 0.2078 won the third place.

Table 4
Results of the AILA Task 2 - Statute Retrieval ranked by MAP

Run_ID	MAP	BPREF	recip_rank	P@10
fs_hit_1_task1b_03	0.2139	0.1587	0.3371	0.13
fs_hit_1_task1b_01	0.1703	0.0945	0.2196	0.12
fs_hit_1_task1b_02	0.1703	0.0945	0.2196	0.12

From the experimental results, the methods in task1b are not achieved good performance, and the result of using Language Model with Dirichlet smoothing is slightly higher.

Table 5
Results of the precedent retrieval task.

Run_ID	Macro Precision	Macro Recall	Macro F-Score	Accuracy
fs_hit_1_3	0.484	0.449	0.41	0.574
fs_hit_1_2	0.456	0.433	0.405	0.578
fs_hit_1_1	0.486	0.406	0.385	0.508

From the experimental results, the results of using the Bert are slightly higher than those using the Logistic Regression method.

4. Conclusions

The paper presents the methods we used in the FIRE2020 AILA. We propose text retrieval methods (BM25 and Language Model) for Task1. Bert model and Logistic Regression method are employed for tackling the Task2. Our ranks in three tasks are in a modest position among all the participants in the score board. In future we will continue study the retrieval model and classification model that we used now, and find other methods or models like Bert-large model.

5. Acknowledgements

This work is supported by National Social Science Fund of China (No.18BYY125)

6. References

[1] Bhattacharya, Paheli and Mehta, Parth and Ghosh, Kripabandhu and Ghosh, Saptarshi and Pal, Arindam and Bhattacharya, Arnab and Majumder, Prasenjit, Overview of the FIRE 2020 AILA track: Artificial Intelligence for Legal Assistance. Proceedings of FIRE 2020 - Forum for Information Retrieval Evaluation. Hyderabad, India, December, 2020.

[2] Robertson, S., Steve Walker, and H. BM. "GM Okapi at trec-3." Proceedings of the Third Text REtrieval Conference (TREC 1994). 1994.

[3] Zhai, Chengxiang, and John Lafferty. "A study of smoothing methods for language models applied to ad hoc information retrieval." ACM SIGIR Forum. Vol. 51. No. 2. New York, NY, USA: ACM, 2017.

[4] Devlin J, Chang M W, Lee K, et al. Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805, 2018.