The Language Model for Legal Retrieval and Bert-based Model for Rhetorical Role Labeling for Legal Judgments

Yujie Xu[b], Tang Li[b], Zhongyuan Han[a,*]

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

Abstract

This paper mainly introduces the solutions to the two tasks published in FIRE2020(forum for information retrieval evaluation), For Task1 (statistic retrieval), The task 1 is, for a given query(description of a situation), identify relevant statutes and prior-cases. This task includes two subtasks, Task1a (identifying relevant prior cases) and Task1b (identifying relevant statistics), For these two subtasks, we use the language model to score each query, and then rank them according to the score. For Task2(rhetorical role labeling for legal judgments), It requires us to classify sentences. We think it's a multi-classification problem, and finally, we use Bert to complete the classification task. In the final result, the score of Task1a is 0.125, the score of Task1b is 0.2003, and the accuracy of Task2 is 0.549. The results and experiments show that the language model is a better way to complete Task1 and Bert is better to complete task2.

Keywords

Legal Retrieval, Rhetorical Role Labeling, Language Model, Bert,

1. Introduction

With the gradual maturity of the social legal system, laws and regulations have become more detailed and standardized, and people's demand for legal aid is gradually increasing. Compared with the low efficiency of artificial legal aid, a series of advantages such as high efficiency and high accuracy of artificial intelligence legal aid is gradually highlighted.

In this regard, FIRE2020 proposed a task and named it AILA2020 (artificial intelligence for legal assistance) to improve the legal aid of artificial intelligence, For the two subtasks in Task1, they provided 10 short descriptions of a legal situation, 3000 judgments delivered by the Supreme Court of India and 197 statutes (Sections of Acts) from Indian law. Retrieve the most relevant case documents or statements for a given query. For Task 2 they provide 8096 rhetorical sentences as training data and 1905 test data, Among them, 8096 training data sentences are classified into one of the following seven semantic segments / rhetorical roles, They are Fact, Ruling by Lower Court, Argument, Statute, Precedent, Ratio of the decision and Ruling by Present Court We are required to divide 1905 test data into these seven categories.

2. Methods

2.1 Methods for Task1a

Fig. 1 describes our method of solving Task1 with the language model.

Forum for Information Retrieval Evaluation 2020, December 16–20, 2020, Hyderabad, India
EMAIL:1520207872xyj@gmail.com (B. 1); itangkk@gmail.com (B. 2); hanzhongyuan@gmail.com (A. 1)(*corresponding author)
ORCID: 0000-0001-8960-9872 (A. 1)


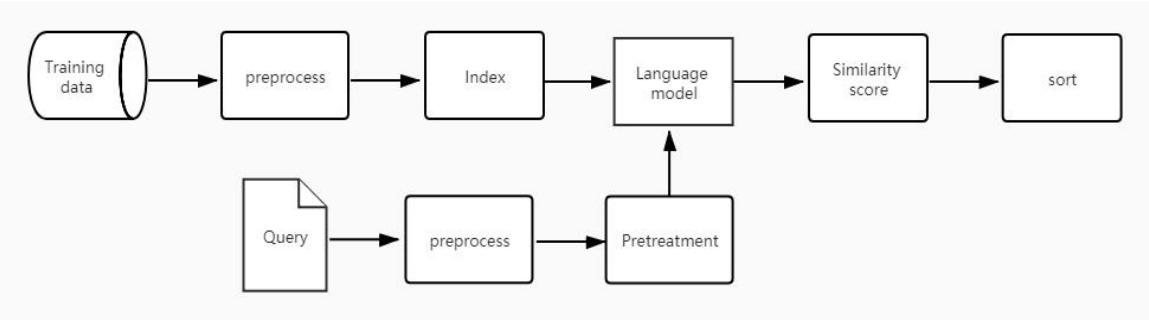

Fig. 1 method of Task1

For Task1a, we tried many models to solve the problem. After many experiments, we finally used Two-Stage Language Model to solve the problem.

After getting the data, we remove the keywords such as period, comma and semicolon from all queries and query documents. At the same time, we also remove some common words to reduce the impact of these words on the results.

For Two-Stage language Model, the document language model is effectively smoothed in two steps. In the first stage, the document language model is smoothed using a Dirichlet prior with the collection language model as the reference model. In the second stage, the smoothed document language model is further interpolated with a query background language model[1]. We used Indri[2] tool to index the document. In the subsequent retrieval, we used Two-Stage Language Model to calculate the similarity between query and document. The similarity is computed using Eq.(1)[1]. When we get the similarity between each query and document, we sort it. The higher the score, the higher the ranking, the more similar the query and document. After many experiments, we found that when $\mu = 2500$, $\lambda = 0.8$, the retrieval results of Task1a is the best.

$$p(q \mid \hat{\theta}_D, \lambda, u) = \prod_{i=1}^{m}((1-\lambda)p(q_i \mid \hat{\theta}_D) + \lambda p(q_i \mid u)) = \prod_{i=1}^{m}((1-\lambda)\frac{c(q_i,d) + \mu p(q_i \mid S)}{\mid d \mid + \mu} + \lambda p(q_i \mid u)) \quad (1)$$

2.2 Methods for Task1b

For Task1b, we not only choose the method of Task1a, but also choose the Jelinek-Mercer language model[2] to calculate the similarity between query and document. Eq.(2) is used to calculate the similarity between query and document. Before retrieval, we also process the given data by word-based n-gram and character-based n-gram. We find that the performance of character-based n-gram is much better than that of word-based n-gram, while the n-gram based on 2-7 achieves the best result.

$$p(w \mid \hat{\theta}_D) = \lambda_D p_{ML}(w \mid \hat{\theta}_D) + (1-\lambda_D)p(w \mid \hat{\theta}_C) \quad (2)$$

2.3 Methods for Task2

For Task2, we think that this is a multi-classification problem. We use the Logistic Regression Model and lighter version of Bert[3]. The weight of bert is set with uncased_ L-12_ H-768_ A-12. 8096 training data without any processing are used to fine-tuning the Bert model with the parameters(max-Len = 124, batch_ Size = 24, units = 7, epoch = 2).

3. Experimental Setting

[2] http://www.lemurproject.org/

[3] https://github.com/bojone/bert4keras

3.1 Parameter Selection

For Task1a, we tried to take different values of μ and λ of Two-Stage Language Model to observe their effects. In fig. 2, we take the results of different λ when $\mu = 1500$ and $\mu = 2500$.

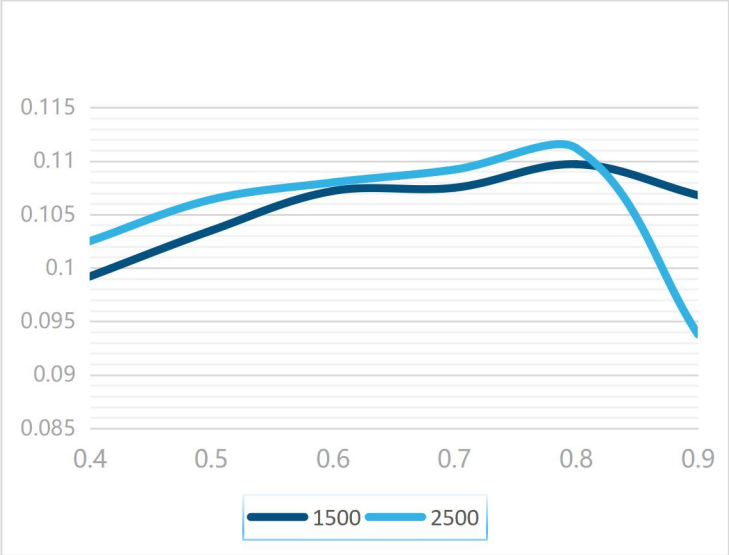

Fig. 2
Experimental results of different parameter combinations in Task1a

For Task1b, we tried to take different values of μ and λ of Two-Stage Language Model to observe their effects. In Fig. 3, we take the results of different λ when $\mu = 1500$ and $\mu = 2500$.

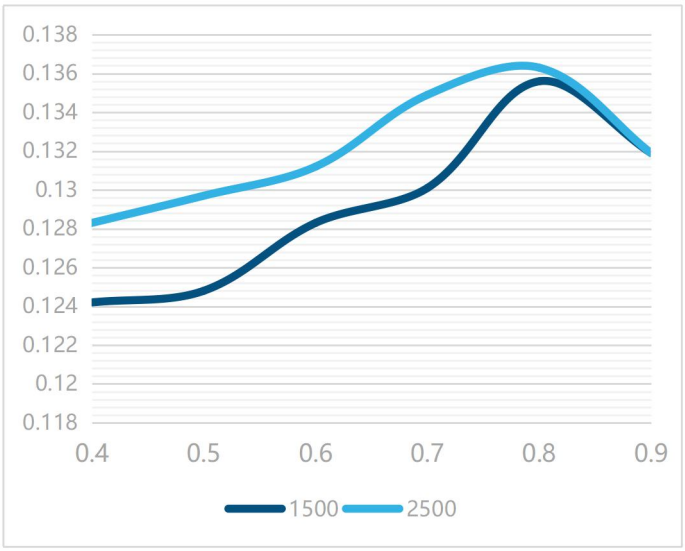

Fig. 3
Experimental results of different parameter combinations in Task1b

In Task1b, we tried to select different n-gram processing to observe their effects. The experimental results are shown in Table 1.

In conclusion, $\mu = 2500$, $\lambda = 0.8$ can achieve better results. In Task1b processing, character level 2 + 3 + 4 + 5 + 6 + 7 has higher accuracy than other results.

Table 1
Experimental results of different n-gram processing combinations in Task1b

NO.	N-gram processing	Map
1	Char-2gram	0.0728
2	Char-2+3gram	0.1197
3	Char-2+3+4gram	0.1329
4	Char-2+3+4+5gram	0.1360
5	Char-2+3+4+5+6gram	0.1368
6	Char-2+3+4+5+6+7gram	0.1473
7	Char-2+3+4+5+6+7+8gram	0.1436
8	Char-2+3+4+5+6+7+8+9gram	0.1443
9	Char-2+3+4+5+6+7+8+9+10gram	0.1434
10	Char-2+3+4+5+6+7+8+9+10+11gram	0.1343
12	Char-2+3+4+5+6+7+8+9+10+11+12gram	0.1325

3.2 Experimental Results

For Task 1, we submitted three groups of results. Table 2 and Table 3 are the experimental results of the test data we submitted[3].

Table 2
The performance of our submitted results for Task1a

Run_ID	MAP	BPREF	recip_rank	P @ 10
fs_hit_2_task1a_01	0.125	0.0724	0.1906	0.07
fs_hit_2_task1a_02	0.0126	0	0.041	0.02
fs_hit_2_task1a_03	0.0123	0	0.0395	0.02

Table 3
The performance of our submitted results for Task1b

Run_ID	MAP	BPREF	recip_rank	P @ 10
fs_hit_2_task1b_01	0.2003	0.1587	0.3452	0.1
fs_hit_2_task1b_02	0.1777	0.1247	0.2546	0.12
fs_hit_2_task1b_03	0.1886	0.132	0.279	0.1

For Task 2, we submitted two sets of results. Table 4 shows the experimental results of the test data we submitted.

Table 4
The performance of our submitted results for Task2

Run_ID	Precision	Recall	F-Score	Accuracy
fs_hit2_1	0.411	0.465	0.405	0.535
fs_hit2_2	0.455	0.427	0.398	0.549

4. Conclusions

This paper introduces the evaluation method we used in FIRE2020 AILA. Compared with other results, we have exposed many deficiencies. For the task of identifying related prior cases, the final evaluation results show that BM25 and TF-IDF are better than our methods. while for multi-classification tasks, Bert shows good results.

5. Acknowledgements

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

6. References

[1] ChengXiang Zhai, John Lafferty, "Two-Stage Language Models for Information Retrieval". The Twenty-Fifth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval.

[2] Guodong Ding, Bin Wang, " GJM-2: A Special Case of General Jelinek-Mercer Smoothing Method for Language Modeling Approach to Ad Hoc IR". Information Retrieval Technology, Second Asia Information Retrieval Symposium, AIRS 2005, Jeju Island, Korea, October 13-15, 2005, Proceedings.

[3] 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