AI-lead Court Debate Case Investigation

Changzhen Ji, Conghui Zhu, Tiejun Zhao
Harbin Institute of Technology
Harbin, Heilongjiang, China
czji_hit@outlook.com,conghui@hit.edu.cn,tjzhao@hit.edu.cn

ABSTRACT

The multi-role judicial debate composed of the plaintiff, defendant, and judge is an important part of the judicial trial. Different from other types of dialogue, questions are raised by the judge, The plaintiff, plaintiff's agent defendant, and defendant's agent would be to debating so that the trial can proceed in an orderly manner. Question generation is an important task in Natural Language Generation. In the judicial trial, it can help the judge raise efficient questions so that the judge has a clearer understanding of the case. In this work, we propose an innovative end-to-end question generation model-Trial Brain Model (**TBM**) to build a Trial Brain, it can generate the questions the judge wants to ask through the historical dialogue between the plaintiff and the defendant. Unlike prior efforts natural language generation, our model can learn the judge's questioning intention through predefined knowledge. We do experiments on real-world datasets, the experimental results show that our model can provide a more accurate question in the multi-role court debate scene.

CCS CONCEPTS

• **Computer systems organization** → **Embedded systems**; *Redundancy*; Robotics; • **Networks** → Network reliability.

KEYWORDS

Natural Language Generation, multi-role, Trial Brain

Table 1: Example Dialog in Court Debate Dataset

Role	**Dialogue**
…	…
Judge	Defendant, is there any evidence to provide to the court?
Defendant	No.
Judge	Plaintiff, what's your relationship with <person>?
Plaintiff	Friends.
Judge	What did <person> borrow money for at that time?
Plaintiff	To operate a supermarket.
Defendant	I heard the people say they were in the same circle.
Judge	Which circle?
Defendant	Gambling circle.
Judge	Plaintiff, do you know whether <person> participated in gambling?
Plaintiff	I don't know. I'm not with him.
Judge	When <person> borrowed money from you, did you agree that it was his personal debt?
Plaintiff	No.
Judge	Defendant, who paid your living expenses with <person>?
Defendant	It is my expenditure.
Judge	Who pays for the family expenses?
Defendant	Me.
…	…

1 INTRODUCTION

The contradiction between the gradual increase of people's demands in pursuing social justice and relatively scarce public resources is one of the prominent contradictions in the current society. In a legal context, a lengthy and expertise-demanding trial can be a high threshold for a litigant, while the judge has to spend significant efforts to investigate the case and explore exhaustive questionable factors. This can be very challenging for junor judges, while a careless negligence can bring unforgivable consequences. Unfortunately, federal/district court judges are experiencing daunting workload, e.g., statistics show that the typical active federal district court judge closed around 250 cases in a year [1, 2]. Applying novel artificial legal intelligence techniques to facilitate the lawsuit process so as to alleviate the information overload for judges is of great significance.

In a longer term, this pioneer investigation can provide critical potentials to enable automatic trail processing system so as to improve trial efficiency. For junior judges, such automation accompanying can provide important assistance during an ongoing trial debate or court rehearsal.

Unlike existing one-to-one dialogue systems, court debate is conducted under multi-role scenario, and an experienced judge may propose the next question with respect to the following factors. First, the prior debate context can be important to decide the content of the forthcoming question (algorithmically, debate context characterization can be essential for this task). Second, response's role information is of significant, i.e., same content from different parties can deliver very different information (algorithmically, role information should be integrated into the debate context representation learning). Third, judge's intention can scope the content of

LegalAI'20@SIGIR'20, July 30, 2020, Virtual Event, China

ACM ISBN TBA…$15.00
https://doi.org/TBA

the forthcoming question. For instance, the judge may raise several related question (e.g., when did you get divorced?, who raise the child?, who cover the daily household expense?) in te- rms of the current intent (e.g., to confirm if the spouse should be also responsible for the debt). After the judge collected enough information of the current intent, he/she can switch to the next intent (e.g., to check if there is actual agreement on the term of the loan).

we propose an innovative end-to-end question generation model to build a trial brain, it can generate the questions the judge wants to ask through the historical dialogue between the plaintiff and the defendant. Unlike prior efforts natural language generation, our model can learn the judge's questioning intention through predefined knowledge. Fig. 1 depicts the systematic structure of the proposed model.

To sum up, our contributions are as follows:

(1) The proposed model is able to learn the judge intention transition for question generation navigation through predefined knowledge.
(2) The proposed model can provide a more accurate question in the multi-role court debate scene.

2 PROBLEM FORMULATION

Let D denote an arbitrary dialogue fragment, containing L utterances. Each utterance U_i in D is composed of a sequence of l words (namely sentence) S_i along with the associated role (of the speaker) r_i. We define the last question in D raised by the judge as U^q and its historical conversations is denoted as $D^- = \{U_1, U_2, ..., U_n\}$. In the task of question generation, the proposed algorithm can generate U^q given a corresponding D^-.⊠

To be clarified, the definition of important notations in the following sections are illustrated as follows:

- D: a debate dialogue fragment containing L utterances;
- r_i: the role of the speaker in U_i (i.e. judge, plaintiff, defendant and witness);
- S_i: the text content of U_i;
- S^q: the text content of U^q;
- D^-: the historical conversations of U^q;
- I_i: the intent of utterance U_i;

Note that $\mathrm{U_i}$, $\mathrm{r_i}$, $\mathrm{S_i}$, and $\mathrm{I_i}$ represent the embedding representations of the corresponding variables in the list.

3 MODEL

The innovative multi-view utterance encoder characterizes three kinds of information - role embedding, semantic embedding, and legal knowledge embedding. More importantly, intention, as an important latent variable, navigates the direction of question generation, which is coached by LKG knowledge transformation. Finally, we use pointer generator networks [9] further enhance the quality of question generation.

3.1 Multi-Role Dialogue Encoding

3.1.1 Role Representation. In term of role embedding, we use dense vectors to represent different roles (e.g., presiding judge, plaintiff, defendant and witness) in the debate dialogue.

3.1.2 Semantic Representation.

- *Utterance Layer.* In the utterance layer, we utilize a Bidirectional Long-Short Term Memory networks (Bi-LSTM) [4] to encode the semantics of the utterance while maintaining its syntactics.

- *Dialogue Layer.* To represent the global context in a dialogue, we use another Bi-LSTM to encode the dependency between utterances to obtain a global representation of an utterance as dialogue representation, denoted as X.

3.1.3 Legal Knowledge Representation. Legal knowledge is an element marked by the judge, such as (borrowing time, loan amount.....). We also use dense vectors to represent different element, and then encode it by LSTM, expressed as $\overline{\mathrm{h^p}}$.

3.2 Representation Enhancement via Intent Navigation

When judge construct the forthcoming question, he/she should consider three kinds of information: (1) the intent of the question, (2) the content of the question, and (3) the litigant role being asked. It motivates us to learn the intent transition to navigate the forthcoming question generation. To represent the intent of judge, we rely on the Legal Knowledge to learn the navigation among different sequence of legal concepts (see Eq. 1). At the same time, we learn a role transfer matrix to represent the role to be asked, in other words, the role who answer the generated question (see Eq. 2). Note that the role used in Sec. 3.1.1 is speaker's role of the corresponding utterance while the role mentioned here is the responser's role of the current utterance[1]

$$\mathrm{I} = \sigma(k^I * \overline{\mathrm{h^p}}) \tag{1}$$

$$\mathrm{R} = \sigma(k^R * \mathrm{r}) \tag{2}$$

The two parameters k^I and k^R stands for the learnable hidden matrix for simulating the intent transfer and the response-role transfer respectively, in which the matrix elements are all values between 0 and 1.

We merge intention information and next role information as below:

$$\mathrm{H} = ([\mathrm{I_1}, \mathrm{R_2}], [\mathrm{I_2}, \mathrm{R_3}], ..., [\mathrm{I_i}, \mathrm{R_{i+1}}]) \tag{3}$$

where $\mathrm{R_{i+1}}$ represents the next role of the current utterance.

We further compress the original redundant information and assign more weight to important information via attention mechanism:

$$\mathrm{Y} = \sum_{i=1}^{n} \frac{exp(\mathrm{I_i})}{\sum_{i=1}^{n} exp(\mathrm{I_i})} * \mathrm{H} \tag{4}$$

Next, we fuse the original information with the intent/role transformation information:

$$\mathrm{Z} = [\mathrm{X}, \mathrm{Y}, \mathrm{X} * \mathrm{Y}, \mathrm{X} - \mathrm{Y}] \tag{5}$$

[1] We mainly focus on the response-role of judge's question since commonly the next role of litigant's answer will be always the judge.

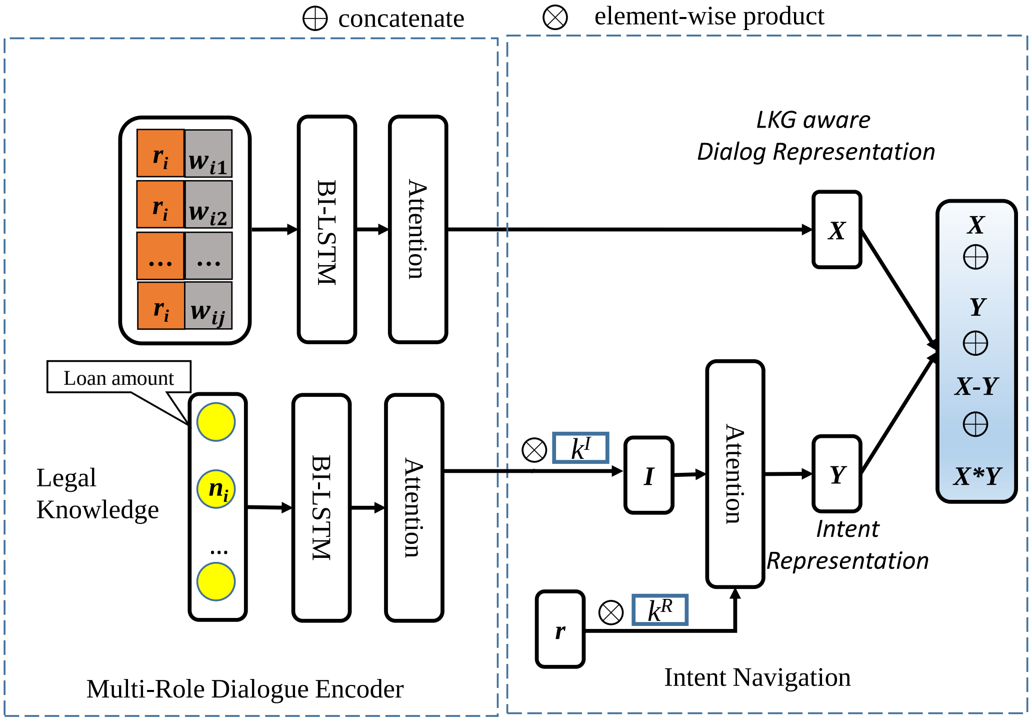

Figure 1: Network Architecture of the Proposed Method

3.3 Parameter Optimization

In question generation learning, for each dialog D, we use cross-entropy to formulate the problem as follows:

$$\begin{aligned} loss &= -\log P(S^q \mid D) \\ &= -\sum_{j=1}^{l} \log P(w_{ij} \mid w_{i1:j-1}, D) \end{aligned}$$

Denoting all the parameters in our model as δ. Therefore, we obtain the following optimized objective function:

$$\min_{\theta} loss = loss + \lambda \, ||\delta||_2^2 \tag{6}$$

To minimize the objective function, we use the diagonal variant of Adam in [11]. At time step t, the parameter δ is updated as follows:

$$\delta_t \leftarrow \delta_{t-1} - \frac{\mu}{\sqrt{\sum_{i=1}^{t} f_i^2}} f_t \tag{7}$$

where μ is the initial learning rate and f_t is the sub-gradient at time t.

According to the evaluation results on the development set, all the hyperparameters are optimized on the training set.

4 EXPERIMENTAL SETTINGS

4.1 Dataset

In the experiment, we collected 136,019 court debate records of civil Private Loan Disputes cases, from which we randomly extracted[2] 302,650 continuous dialogue fragments as independent samples for training, developing and testing[3]. In total, it contains more than 4 million sentences and each dialogue fragment, on average, contains 13.38 sentences. The details of the dataset is illustrated in Table 2.

Table 2: Statistics of the Processed Dialogue Fragments

Dataset	#Samples	#Utterances	#avg_length
train	242,120	3,238,956	13.38
test	30,265	404,833	13.38
development	30,265	404,870	13.38
Total	**302,650**	**4,048,659**	**13.38**

4.2 Baselines

In order to demonstrate the validity of our model, we selected some traditional classical methods and the latest mainstream methods for text generation. The tested baselines are illustrated as follows:

[2] We only selected the fragments in which there are at least five historical utterances as context for our task of next question generation.

[3] The entire dataset is divided by a ratio of 8:1:1 for training, developing and testing, respectively.

Table 3: Main Results of All Test Methods. Note that the results show in TBM(our) rows are statistically significant different from the corresponding value of all the baseline models (p-value< 0.001).

Model	R.-1	R.-2	R.-3	R.-L	BLEU
LSTM	29.33	14.97	10.34	26.65	11.59
ByteNet	35.57	19.47	14.27	32.56	17.73
ConvS2S	36.35	20.42	15.98	33.03	17.97
S2S+attention	36.54	20.72	16.40	33.29	17.96
PGN	37.67	21.93	17.42	34.39	18.75
Transformer	37.59	23.26	18.71	35.38	18.58
TBM(our)	**39.02**	**24.56**	**21.03**	**38.12**	**24.17**

- **LSTM**[4]: We replace all bidirectional LSTM with LSTM in our proposed model.
- **ConvS2S** [3]: LSTM be replaced CNN in the encoder.
- **ByteNet** [5]: It is a one-dimensional convolutional neural network that is composed of two parts, one to encode the source sequence and the other to decode the target sequence.
- **S2S+attention** [7]: The Seq2Seq framework relies on the encoder-decoder paradigm. The encoder encodes the input sequence, while the decoder produces the target sequence. Attention mechanism is added to force the model to learn to focus on specific parts of the input sequence when decoding.
- **PGN** [9]: It is another commonly used framework for text generation which enables copy mechanism to aid accurate reproduction of information, while retaining the ability to produce novel words through the generator.
- **Transformer** [10]: A neural network architecture based on self-attention mechanism.

4.3 Evaluation Metrics

To automatically assess the quality of the *generated question*, we used ROUGE [6] and BLEU [8] scores to compare different models. We report ROUGE-1, ROUGE-2, ROUGE-3 as the means of assessing informativeness and ROUGE-L as well as BLEU-4 for assessing fluency.

5 RESULT DISCUSSION

The performance of all tested methods is reported in Table 3. As the upper part of Table 3 shows, the proposed method **TBM** is significantly (p-value< 0.001) superior than all the tested baselines over all the evaluation metrics.

6 CONCLUSION AND OUTLOOK

Dialogue generation has been well studied in NLP. At present, some achievements have been made. However, it is also faced with great challenges. Many superior models only achieve good results in specific field. In this paper, we define a new task of the question generation in judicial trial. The proposed Trial Brain Model can learn the judge's questioning intention through predefined knowledge. Judge-centered debate context heterogeneity is the landmark of this model, i.e., a delicately designed multi-role dialogue encoding mechanism via Legal Knowledge with the representation enhancement through intent navigation by simulating the intention switch across different conversations. The empirical findings validate the hypothesis of this bionic design for judge logic reduction. An extensive set of experiments with a large civil trial dataset shows that the proposed model can generate more accurate and readable questions against several alternatives in the multi-role court debate scene.

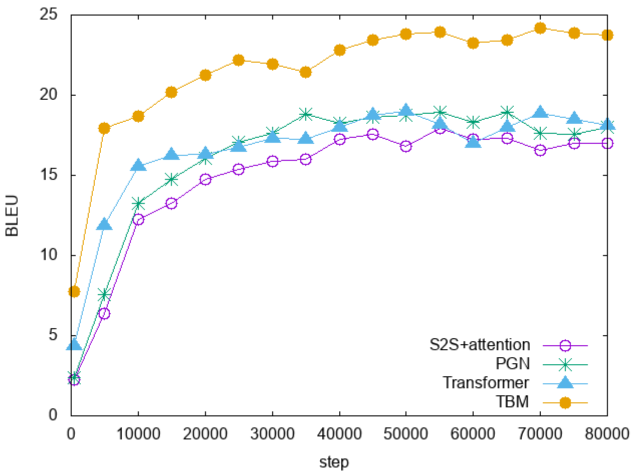

Figure 2: The performance of tested methods.

7 ACKNOWLEDGMENTS

This work is supported by National Key R&D Program of China (2018YFC0830200;2018YFC0830206).

REFERENCES

[1] Judge Information Center. [n. d.]. As Workloads Rise in Federal Courts, Judge Counts Remain Flat. available from https://trac.syr.edu/tracreports/judge/364/. Accessed: 2019-05-06.

[2] Judge Information Center. [n. d.]. Some Federal Judges Handle Inordinate Caseloads. available from https://trac.syr.edu/tracreports/judge/501/. Accessed: 2019-05-06.

[3] Jonas Gehring, Michael Auli, David Grangier, Denis Yarats, and Yann N Dauphin. 2017. Convolutional sequence to sequence learning. JMLR. org, 1243–1252.

[4] Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation* 9, 8 (1997), 1735–1780.

[5] Nal Kalchbrenner, Lasse Espeholt, Karen Simonyan, Aaron van den Oord, Alex Graves, and Koray Kavukcuoglu. 2016. Neural machine translation in linear time. *arXiv preprint arXiv:1610.10099* (2016).

[6] Chin-Yew Lin and Eduard Hovy. 2003. Automatic evaluation of summaries using n-gram co-occurrence statistics.

[7] Ramesh Nallapati, Bowen Zhou, Caglar Gulcehre, Bing Xiang, et al. 2016. Abstractive text summarization using sequence-to-sequence rnns and beyond. *arXiv preprint arXiv:1602.06023* (2016).

[8] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: a method for automatic evaluation of machine translation. Association for Computational Linguistics, 311–318.

[9] Abigail See, Peter J Liu, and Christopher D Manning. 2017. Get to the point: Summarization with pointer-generator networks. *arXiv preprint arXiv:1704.04368* (2017).

[10] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. 5998–6008.

[11] Matthew D Zeiler. 2012. ADADELTA: an adaptive learning rate method. *arXiv preprint arXiv:1212.5701* (2012).