Disentanglement-Augmented Rationale Extraction

Anonymous Author(s)
Affiliation
Address
email

Abstract

Rationale extraction can be considered as a straightforward method of improving the model explainability, where rationales are a subsequence of the original inputs, and can be extracted to support the prediction results. Existing methods are mainly cascaded with the *selector* which extracts the rationale tokens, and the *predictor* which makes the prediction based on selected tokens. Since previous works fail to fully exploit the original input, where the information of non-selected tokens is ignored, in this paper, we propose a Disentanglement-Augmented Rationale Extraction (DARE) method, which encapsulates more information from the input to extract rationales. Specifically, it first disentangles the input into the rationale representations and the non-rationale ones, and then learns more comprehensive rationale representations for extracting by minimizing the mutual information (MI) between the two disentangled representations. Besides, to improve the performance of MI minimization, we develop a new MI estimator by exploring existing MI estimation methods. Extensive experimental results on two real-world datasets and simulation studies clearly validate the effectiveness of our proposed method.

1 Introduction

Although the performance of deep neural networks (DNNs) has significantly improved across a range of natural language understanding tasks, the inability of DNNs to provide the explainability for their predictions still remains a serious risk. To this end, several researchers [29, 21, 23] have focused on improving the explainability of DNNs. Among them, the rationale extraction method [20, 3] can provide an intuitive explanation by identifying the important features of inputs. Specifically, it extracts a short and coherent part of original inputs (i.e., the *rationale*) as an explanation to support the prediction results when yielding them.

Traditional rationale extraction approaches [20, 3, 36, 27] cascade the *selector* and the *predictor*. As shown in Figure 1(a), we illustrate this type of approaches with an example of the charge prediction, where charges are automatically predicted based on the case fact. Specifically, the *selector* first selects a subsequence of the fact description (i.e., the bolded tokens in Figure 1(a). Then, the *predictor* yields the charge result based on the selected tokens. And the extracted subsequence is defined as the rationale. However, since the extractive rationales only depend on the comparison between the prediction results and labels, this kind of approach may fail to fully exploit the information of the original input. To solve that, Sha et al. [31] add an external *guider* (i.e., Figure 1(b)) which takes the whole text as the input and generate the accurate but uninterpretable representations as most DNNs do to predict the result. Then, the *guider* utilizes the above representations to guide the *predictor* to yield more comprehensive task-related representations with an adversarial-based method. Since this method fails to utilize the information of the original text, where the non-rationale tokens are ignored, we argue that this "*guidance pattern*" can be further explored to improve the rationale extraction.

Submitted to 36th Conference on Neural Information Processing Systems (NeurIPS 2022). Do not distribute.

Along this research line, in this paper, we propose a " *self-guided* " method, **D**isentanglement-**A**ugmented **R**ationale **E**xtraction (DARE), which augments the traditional rationale extraction approaches with the disentangled representations learning. Different from the previous model that requires external guidance, the DARE aims to guide itself to extract more comprehensive rationales by squeezing more information from the input. Specifically, we first disentangle the original inputs into two parts: the rationale tokens and the non-rationale ones through the *selector*, where the two type tokens should be independent ideally. Then, in the *predictor* which consists of an encoder and a classifier, these two type tokens are passed into a shared encoder to obtain the corresponding representations. Finally, the DARE utilizes rationale representations to yield the results. Meanwhile, the *guider* in DARE adopts the non-rationale representations to guide the rationale ones to be more comprehensive by reducing the dependency between the two representations (i.e., minimizing their Mutual Information (MI)).

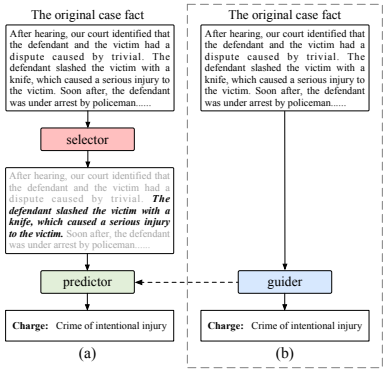

Figure 1: Schematic of existing rationale extraction approaches presented in this paper. (a) illustrates a traditional process of rationale extraction which utilizes the selected rationale (in **bold**) to yield results. (b) shows an external guider to guide the rationale selection based on the whole input.

Besides, to improve the performance of minimizing MI, based on the state-of-the-art MI minimization method, Contrastive Log-ratio Upper Bound (CLUB) [10, 11], we propose a new implementation of CLUB (i.e., CLUB_NCE) by exploring the relationship between CLUB and InfoNCE (a classical maximizing MI method [26]). Following the experimental setup of [28, 10], we evaluate the effectiveness of CLUB_NCE on two simulation studies by comparing with other MI minimization methods.

To validate the effectiveness of our proposed DARE, we perform extensive experiments on the BeerAdvocate dataset [24] which is utilized to the multi-aspect sentiment analysis, and a legal judgment prediction dataset (CAIL2018) [34] which is adopted to predict judgment results including charges, law articles and terms of penalty. The experimental results empirically show that DARE has achieved better performance on both extracting rationales and predicting task results compared to other state-of-the-art rationale extraction methods.

2 Disentanglement-Augmented Rationale Extraction

2.1 Problem Definition

Given a text input $x = \{x_1, x_2, \ldots, x_n\}$ consisting of n tokens and the ground truth y, our goal is first to learn a mask variable $m = \{m_1, m_2, \ldots, m_n\}$, and then learn a model which adopts the rationale $z = m \odot x = \{m_1 \cdot x_1, m_2 \cdot x_2, \ldots, m_n \cdot x_n\}$ to yield the prediction results. Take the case in Figure 1 for example, based on the fact description x, DARE aims to yield the charge result y while extracting the rationale z to support this result. Next, we will present the details of our proposed " *self-guided* " method, DARE, which consists of the *selector*, *predictor* and *guider*.

2.2 Architecture of DARE

To utilize the non-rationale representations to enhance the rationale ones, and further extract more accurate rationales, we propose the DARE method which is shown in Figure 2. In DARE, we first utilize the *selector* to select the rationale tokens and the non-rationale ones, and adopt the shared encoder to generate the corresponding representations. Then, the classifier in the *predictor* can yield the prediction results based on the rationale representations. Meanwhile, the *guider* considered as a regularizer minimizes the MI between the above two types of representations to ensure they are explicitly disentangled, which can make the rationale representations more comprehensive.

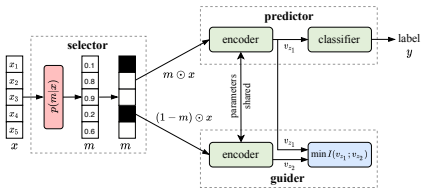

Figure 2: Architecture of DARE consisting of the selector, predictor and guider : the white boxes indicate the rationales related tokens and the black are non-related ones.

2.2.1 Selector

To select the rationale tokens, the *selector* takes $x = \{x_1, x_2, \ldots, x_n\}$ as the input to generate a binary variable $m = \{m_1, m_2, \ldots, m_n\}$, and adopts m to sample the rationale and non-rationale tokens. Specifically, we first utilize the *selector* to generate a probability distribution $p(m|x)$ which represents the probability of sampling each x_i as the part of the rationale. To ensure this operation of sampling is differentiable, we adopt HardKuma reparameterization trick [3] to generate the $p(m|x)$:

$$k_j = F^{-1}_{Kuma}(u_j; a_j, b_j), t_j = l + (r - l) \times k_j, p(m_j|x) = \min\left(1, \max\left(0, t_j\right)\right), \tag{1}$$

where u_j is random sampled from the uniform distribution $U(0, 1)$, $F^{-1}_{Kuma}(u; a, b) = \left(1 - (1 - u)^{1/b}\right)^{1/a}$ is the inverse c.d.f. of the Kumaraswamy distribution [19], a and b are parameterized by DNNs which takes x as the input, and r and l are fixed hyperparameters. Finally, we can derive the rationale tokens as $z_1 = m \odot x$ and the non-rationale ones as $z_2 = (1 - m) \odot x$.

2.2.2 Predictor

The *predictor* $q_\psi(y|z_1)$ which consists of an encoder and a classifier outputs the prediction results based on the rationale tokens z_1. Specifically, we first re-embed the z_1 into continuous hidden states by the encoder to obtain the rationale representations $v_{z_1} \in \mathbb{R}^d$, where d is the hidden size. Then, we pass the v_{z_1} into the classifier to yield the results y which is predicted based exclusively on the rationale tokens extracted by *selector*, and the task loss function can be formulated as :

$$\mathcal{L}_{task} = \mathbb{E}_{z_1 \sim p_\sigma(z_1|x)}\left[-\log q_\psi(y \mid z_1)\right], \tag{2}$$

where p_σ is the neural network of *selector*.

2.2.3 Guider

While predicting the results, to guide the model itself to extract more comprehensive rationales, we augment the disentangled representations learning (DRL) into the *guider*. Specifically, we pass the non-rationale tokens z_2 to the shared encoder described in section 2.2.2 to obtain the non-rationale representations v_{z_2}. Then, we hope v_{z_1} and v_{z_2} should be independent ideally so that the v_{z_1} can contain rich rationale information. To achieve this goal, we propose the mutual information minimization method which reduces the dependency between v_{z_1} and v_{z_2} to disentangle them (i.e., min $I(v_{z_1}; v_{z_2})$). Besides, following the application of MI in DRL [11, 30, 17], we should maximize $I(v_x; v_{z_1}, v_{z_2})$ to ensure v_{z_1} and v_{z_2} encapsulate information from the input x sufficiently, where v_x is the original input representations. However, since v_{z_1} and v_{z_2} are obtained from the separate input, we argue that v_{z_1} and v_{z_2} have preserved the original input itself, and can remove this MI maximization regularization. In section 2.3, we will describe the MI minimization method in detail.

2.3 Mutual Information Minimization

In this section, we first review some concepts of MI, and then present a new MI minimization method (i.e., CLUB_NCE) by discussing the relationships between InfoNCE and CLUB.

2.3.1 Mutual Information

In probability theory and information theory, the mutual information (MI) of two random variables (e.g., X and Y) is a measure of the mutual dependence between the two variables $I(X; Y) = \mathbb{E}_{p(x,y)}\left[\log \frac{p(x,y)}{p(x)p(y)}\right]$. However, directly computing MI values is difficult [4], especially when X and Y are continuous and high-dimensional. To approximate MI, for MI maximization tasks, Oord et al. [26] derive a lower-bound MI estimation (InfoNCE):

$$I_{nce} = \frac{1}{N}\sum_{i=1}^{N} \log \frac{e^{f(x_i, y_i)}}{\frac{1}{N}\sum_{j=1}^{N} e^{f(x_i, y_j)}} = \frac{1}{N}\sum_{i=1}^{N} f\left(x_i, y_i\right) - \frac{1}{N}\sum_{i=1}^{N}\left[\log \frac{1}{N}\sum_{j=1}^{N} e^{f(x_i, y_j)}\right], \tag{3}$$

where $\{(x_i, y_i)\}_{i=1}^{N}$ is a batch of sample pairs and $f\left(x, y\right)$ is a learnable score function. For MI minimization tasks, Cheng et al. [10, 11] propose a Contrastive Log-ratio Upper Bound (CLUB) method:

$$I_{club} = \frac{1}{N}\sum_{i=1}^{N}\log p(y_i|x_i) - \frac{1}{N^2}\sum_{i=1}^{N}\sum_{j=1}^{N}\log p(y_j|x_i), \tag{4}$$

where $p(y|x)$ is a conditional distribution and always parameterized by a Gaussian family[1].

2.3.2 CLUB_NCE

Here, we discuss the relationship between InfoNCE and CLUB. Specifically, by applying the Jensen's inequality in the Eq (3), we can get:

$$I_{nce} \leq \frac{1}{N}\sum_{i=1}^{N} f\left(x_i, y_i\right) - \frac{1}{N^2}\sum_{i=1}^{N}\sum_{j=1}^{N}\log\left[e^{f(x_i,y_j)}\right] = \frac{1}{N}\sum_{i=1}^{N} f\left(x_i, y_i\right) - \frac{1}{N^2}\sum_{i=1}^{N}\sum_{j=1}^{N} f\left(x_i, y_j\right), \tag{5}$$

where we note this inequality is similar to Eq (4) and the main difference is the score function. Interestingly, we find for any critic $f(x, y)$ in InfoNCE, the optimal critic is $f^*(x, y) = \log(p(y \mid x))$ [28] which is equivalent to the critic function in Eq (4).

Inspired by this observation, we propose a new implementation of CLUB (i.e., CLUB_NCE) to minimize MI, which first adopts the trained $f(x, y)$ by InfoNCE to replace the $\log(p(y \mid x))$ in CLUB to calculate the value of I_{club}, and then minimizes it to minimize MI.

We illustrate the process of MI Minimization with CLUB_NCE in Algorithm 1. At each training iteration, we first sample $\{(x_i, y_i)\}_{i=1}^{N}$ from the generator $g_\phi\left(x, y\right)$. Then, we update the $f_\theta\left(x, y\right)$ by maximizing I_{nce}. Next, the value of I_{club} can be calculated as described in Eq (5) and $g_\phi\left(x, y\right)$ can be updated. The neural network f_θ and g_ϕ are trained alternately, which is similar to the training process of GAN [14]. Specifically, in DARE, we set the shared encoder as the g_ϕ, and the *guider* which is parameterized with a full connection layer as the f_θ.

Algorithm 1 MI Minimization with CLUB_NCE

for each training iteration **do**
  Sample $\{(x_i, y_i)\}_{i=1}^{N}$ from $g_\phi\left(x, y\right)$.
  Compute the InfoNCE loss as:
  $I_{nce} = \frac{1}{N}\sum_{i=1}^{N}\log\frac{\exp(f_\theta(x_i,y_i))}{\frac{1}{N}\sum_{j=1}^{N}\exp(f_\theta(x_i,y_j))}$.
  Update the $f_\theta\left(x, y\right)$ by maximizing I_{nce}.
  for $i = 1$; $i <= N$; $i++$ **do**
    $U_i = f_\theta\left(x_i, y_i\right) - \frac{1}{N}\sum_{j=1}^{N} f_\theta\left(x_i, x_j\right)$.
  end for
  Update the $g_\phi\left(x, y\right)$ by minimizing :
  $I_{club} = \frac{1}{N}\sum_{i=1}^{N} U_i$.
end for

2.4 Training

During training, following the setup of [3], we first penalize $\mathcal{L}_{sh} = \frac{1}{n}\sum_{j=1}^{n}[1 - p(m_j = 0|x)]$ to ensure the model extracts short rationales, where $p(m_j = 0|x)$ is the probability of the token not being selected. Then, we add a regularizer to encourage coherence of selected tokens[2]:

$$\mathcal{L}_{re} = \frac{1}{n-1}(\sum_{j=1}^{n-1} p(m_j = 0 \mid x)\left(1 - p(m_{j+1} = 0 \mid x)\right) + \left(1 - p(m_j = 0 \mid x)\right) p(m_{j+1} = 0 \mid x). \tag{6}$$

By combining Eq (2), $\mathcal{L}_{sh}$, $\mathcal{L}_{re}$ and the I_{club} calculated by CLUB_NCE between v_{z_1} and v_{z_2}, our final objective of DARE is defined as $\mathcal{L} = \lambda_1\mathcal{L}_{task} + \lambda_2\left|\mathcal{L}_{sh} - l_r\right| + \lambda_3\mathcal{L}_{re} + \lambda_4 I_{club}$, where λ_1 and λ_4 are fixed hyperparameters, λ_2 and λ_3 are Lagrangian multipliers which will be updated while training, and l_r represents that how many tokens DARE will select.

3 Experiments

In this section, we first compare DARE with some baselines on a beer reviews dataset and a legal judgment prediction dataset. Then, to demonstrate the effectiveness of CLUB_NCE, we compare it with some classical MI minimization methods on both simulation studies and real-world datasets.

[1] `https://github.com/Linear95/CLUB`

[2] For more details of $\mathcal{L}_{sh}$, $\mathcal{L}_{re}$ and HardKuma, refer to [3].

Table 1: Precision, Recall and F1 of selected rationales for three aspects. Among them, "% selected" represents the average proportion of selected tokens in the original text.

Methods	Appearance				Smell				Palate			
	Precision	Recall	F1	% selected	Precision	Recall	F1	% selected	Precision	Recall	F1	% selected
Bernoulli	96.3	56.5	71.2	14	95.1	38.2	54.5	7	80.2	53.6	64.3	7
HardKuma	**98.1**	65.1	78.3	13	**96.8**	31.5	47.5	7	**89.8**	48.6	63.1	7
InfoCal_IB	97.3	67.8	79.9	13	94.3	34.5	50.5	7	89.6	51.2	65.2	7
InfoCal(HK)	97.9	71.7	82.8	13	94.8	42.3	58.5	7	89.4	56.9	69.5	7
DARE (L1Out)	91.5	26.7	41.3	13	84.0	38.0	52.3	7	55.4	57.0	56.2	7
DARE (CLUB)	93.7	73.0	82.1	13	90.9	42.9	58.3	7	88.7	54.3	67.4	7
DARE	95.1	**73.5**	**82.9**	13	88.6	**46.8**	**61.2**	7	85.6	**59.0**	**69.9**	7
(std)	±0.2	±0.3	±0.1	-	±0.8	±0.6	±0.6	-	±0.6	±0.5	±0.2	-

3.1 Beer Reviews Prediction

Beer Reviews Prediction is formulated as a multi-aspect sentiment analysis task, which predicts the ratings (on a scale of 0 to 5 stars) for multiple aspects (e.g., smell, palate) based on the beer reviews. We use the BeerAdvocate [24] dataset containing more than 220,000 beer reviews as our dataset. For the fair comparison, we replicate the pre-process of Lei et al.[20] and Sha et al. [31] where the dataset has been divided into three aspects including appearance, smell, and palate. We normalize the ratings to $[0, 1]$ and adopt them as the ground truth for the regression. For testing, we take 994 reviews for three aspects as our test set, where the aspect-related rationales are annotated by human.

3.1.1 Comparison Methods and Experimental Setup

• **Bernoulli** [20] yields a Bernoulli distribution of each token and samples the token as rationales.

• **HardKuma** [3] proposes a HardKuma distribution based on Kumaraswamy distribution [19] for reparameterized gradient estimates while selecting the rationale tokens.

• **InfoCal** [31] introduces an adversarial-based method to guide the selector-predictor model to extract rationales. Since InfoCal proposed a new LM regularizer to encouraging the consecutiveness of rationales, for a fair comparison, we adopt one of the versions of InfoCal (i.e., **InfoCal (HK**[3]**)** in [31]), which applies the same regularizers $\mathcal{L}_{sh}$ and $\mathcal{L}_{re}$ to ensure the rationales are short and coherence.

• **InfoCal_IB** [31] adopts the Information Bottleneck regularizer to manage the trade-off between selecting short rationales and yielding accurate results by removing the adversarial module of InfoCal.

We replicate the setup of [3] where the architecture of *selector* and *encoder* are RCNN and the loss function is the mean-squared error (MSE) loss. Besides, we adopt the precision, recall, and F1-score to evaluate the performance of the selected rationales. Among them, the precision is defined as how many the selected tokens are in annotated rationales, and the recall represents how many annotated rationale tokens are selected. For a fair comparison, we set the l_r as $\{0.13, 0.07, 0.07\}$ in three aspect datasets, respectively, and initialize the epoch as 50, which are both consistent with the previous methods [3, 31]. Several results of baselines in Table 1 and Figure 3 are directly taken from [31].

3.1.2 Experimental Results

To demonstrate the effectiveness of DARE, we compare it with baselines on the beer review prediction task, and experimental results are shown in Table 1. From the results, we can find DARE achieves better performance on the three aspects of BeerAdvocate datasets, especially on recall and F1 scores, which demonstrates that DARE can select more comprehensive rationales. Specifically, compared with the traditional *selector-predictor* models (i.e., Bernoulli, HardKuma and InfoCal_IB), DARE significantly outperforms them, indicating the effectiveness of our method on utilizing the information of non-rationale tokens, where this type tokens are ignored in previous models. As the *selector* and *predictor* in DARE are based on HardKuma, and our model performs better than it, we can conclude that our disentanglement augmented method with MI minimization is effective.

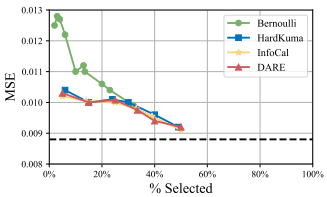

Figure 3: MSE of all aspects on the test set for various percentages of extracted text. The black dashed line represents the full-text baseline which selects all tokens in the input.

[3]HK represents adopting the regularizers (i.e., $\mathcal{L}_{sh}$ and $\mathcal{L}_{re}$) in HardKuma.

Table 2: Judgment prediction results on CAIL. Therein, the underlined scores are the state-of-the-art performances in LJP but lacking explainability, and the results in **bold** performs second only to NeurJudge but with explainability. Results of LJP baselines in Table 2 are quoted from [38].

Methods	Charges				Law Articles				Terms of Penalty			
	Acc	MP	MR	F1	Acc	MP	MR	F1	Acc	MP	MR	F1
FLA	84.72	83.71	73.75	75.04	85.63	83.46	73.83	74.92	35.04	33.91	27.14	24.79
TopJudge	86.48	84.23	78.39	80.15	87.28	85.81	76.25	78.24	38.43	35.67	32.15	31.31
LADAN	88.28	86.36	80.54	82.11	88.78	85.15	79.45	80.97	38.13	34.04	31.22	30.20
NeurJudge	88.89	86.96	85.42	85.73	89.71	86.68	83.92	84.97	41.03	39.52	36.82	36.35
Bernoulli	85.76	82.71	78.04	79.38	86.45	81.94	75.92	77.35	38.19	33.05	32.10	30.32
HardKuma	86.26	86.22	77.83	80.04	85.76	80.92	76.85	77.96	35.94	35.26	27.52	26.09
InfoCal_IB	87.14	84.86	80.70	82.11	87.58	83.55	78.49	80.13	36.60	33.12	28.20	26.82
InfoCal(HK)	87.85	86.79	82.53	83.87	87.65	84.88	80.09	81.67	**38.69**	35.68	31.61	**31.21**
DARE (L1Out)	83.20	80.92	73.74	75.10	85.89	82.76	76.54	78.04	24.09	11.76	9.14	3.72
DARE (CLUB)	82.87	83.19	73.17	74.84	86.48	84.14	77.05	78.83	37.45	33.41	28.99	27.88
DARE	**88.29**	**86.58**	**83.29**	**84.36**	**88.51**	**85.62**	**81.35**	**82.66**	38.43	**36.77**	**33.25**	30.65
(std)	±0.20	±0.37	±0.34	±0.31	±0.28	±0.40	±0.31	±0.38	±0.22	±0.17	±0.41	±0.27

Besides, we analyze the experimental results between DARE and InfoCal(HK) which is also a "guided" model but with an external guider (i.e., Figure 1(b)). DARE does not outperform the InfoCal(HK) on all metrics and some results are close (e.g., F1 in Palate prediction). This is probably because InfoCal(HK) exploits the strength of the external *guider*, a black-box neural network that can yield more accurate but uninterpretable representations. However, as a whole, InfoCal(HK) does not perform as well as our model, which demonstrates the effectiveness of squeezing the non-rationale information from input. We also analyze the effectiveness of CLUB_NCE by comparing other MI minimization methods (e.g., CLUB [10] and L1Out [28]), and the results are shown in section 3.3.3.

Furthermore, we conduct an experiment on predicting the ratings for multiple aspects while adopting the different selected percent of the textual input. As shown in Figure 3, our method DARE achieves a similar performance compared with some baselines on MSE. As the DARE performs better on selecting rationales in Table 1, we can conclude that our model selects more accurate rationales while yielding prediction results.

3.2 Legal Judgment Prediction

Legal Judgment Prediction (LJP) task can be formulated as a text classification problem, which adopts the case fact to yield the judgment results (i.e., charge, law articles, and terms of penalty). We conduct our experiments on publicly available datasets of the **C**hinese **AI** and **L**aw challenge CAIL2018 [34]. CAIL2018 contains criminal cases which consists of fact description and corresponding charges, law articles, and terms of penalty results. Following the data process of Yue et al. [38], we filter out some infrequent and multiple charges cases, and divide the terms into non-overlapping intervals. Then, the LJP task can be formulated as a multi-class classification problem.

3.2.1 Comparison Methods and Experimental Setup

In addition to comparing the methods in the rationale extraction, we also compare with some classical baselines in LJP task including FLA [22], TopJudge [39], LADAN [35] and NeurJudge [38]. The above baselines are trained with a multi-task framework which exploits the dependence among the sub-tasks in LJP (i.e., predicting the charge, law articles, and terms of penalty cooperatively). Besides, we conduct experiments on one of versions of CAIL2018 (denoted by CAIL) which contains 134,739 cases [38]. We set the l_r as $\{0.14, 0.14, 0.14\}$ for training. For evaluating, as the CAIL dataset does not contain annotations of rationales, we first adopt the accuracy (Acc), macro-precision (MP), macro-recall (MR), and macro-F1 (F1) to evaluate the performance of yield judgment results. Then, we provide a qualitative analysis on the rationale extraction. Detailed description of comparison methods and experimental setups can be found in Appendix A.1.

3.2.2 Experimental Results

To evaluate the performance of our model on LJP, we show the experimental results from three aspects. First, comparing with the traditional rationale extraction methods, we can find DARE performs well on LJP tasks, which further validates the effectiveness of our proposed model based on the previous beer reviews prediction experimental results. Meanwhile, the analysis of the comparison between CLUB_NCE and other methods of minimizing MI are shown in section 3.3.3.

HardKuma	The People's Procuratorate alleged that ***the defendant Tommy and Bob forcibly had sexual relations with the victim in the room of hotel***. In this regard, the public prosecution agency cited the following evidence: ***capture history***, ..., and statements and explanations of the defendant. After hearing, our court ***identified that the defendant Tommy used violence and verbal threats with others to forcibly have sexual relations with the victim in the hotel room*** .....
InfoCal(HK)	The People's Procuratorate alleged that the defendant ***Tommy and Bob forcibly had sexual relations with the victim*** in the room of hotel. In this regard, the public prosecution agency cited the following evidence: capture history, ..., and statements and explanations of the defendant. After hearing, our court identified that the defendant Tommy ***used violence*** and verbal threats with others to ***forcibly have sexual relations with the victim in the hotel room*** .....
DARE	The People's Procuratorate alleged that the defendant Tommy and Bob ***forcibly had sexual relations with the victim in*** the room of hotel. In this regard, the public prosecution agency cited the following evidence: capture history, ..., and statements and explanations of the defendant. After hearing, our court identified that the defendant Tommy ***used violence and verbal threats with others to forcibly have sexual relations with the victim in the hotel room*** .....

Figure 4: Visualized selective rationales with different methods. The underlined tokens represent the real rationales which is used to support the practical charge (i.e., rape), and the blue is the predicted rationales. More examples can be found in Appendix A.2.

Then, by comparing with methods in LJP (e.g., NeurJudge), we observe DARE still achieves good overall performance, second only to NeurJudge (the state-of-the-art model in LJP) in most metrics. It is worth noting that these LJP methods are modeled based on the multi-task learning, where the charge, article and term of penalty prediction tasks are related and can enhance each other while DARE is a single-task model (i.e., we need to train a separate model for each task). It demonstrates that DARE can extract effective rationale tokens for prediction. Besides, although these LJP methods achieve promising performance, the judgment results still remain unreliable and difficult to explain. In contrast, DARE can achieve a trade-off between the accuracy and explainability.

As there exist no annotations of rationales in CAIL dataset, we provide a qualitative analysis on rationales extracted by DARE. First, Figure 4 shows an example of the extracted rationale for the charge prediction with different methods, where the underlined tokens represent the practical rationales annotated by human judge. From the observation, DARE can select key rationales which could be adopted to support the prediction results. Besides, comparing with other extracted performance of baselines, DARE can select more rationale tokens, which indicates the strength of utilizing the non-rationale representations.

Furthermore, to verify that DARE has effectively disentangled the textual input into the rationale representations and the non-rationale ones, we visualize these representations spaces using t-SNE [33] in Figure 5. Specifically, we visualize the same case (in Figure 4) predicted with HardKuma and DARE. Figure 5(a) shows the visualization of HardKuma. The blue dashed circle is mostly filled with tokens that were predicted correctly and these tokens are also clustered together, illustrating the effectiveness of HardKuma. The red dashed circle is filled with tokens that HardKuma predicted incorrectly where these tokens are non-rationale tokens, and the positions of these tokens are very close to the blue circle. Figure 5(b) shows the results of the DARE visualization. Among them, the tokens circled by the red dashed is the same as the tokens circled by the red dashed circle in Figure 5(a). It can be seen that after the disentanglement operation, these tokens can be obviously separated and be farther away from the tokens circled in blue. Moreover, these red dashed circled tokens in Figure 5(b) are accurately predicted to be non-rationale tokens. The above visualizations demonstrate that DARE can well disentangle rationale representations.

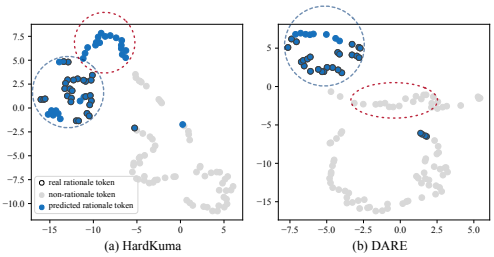

Figure 5: Disentanglement visualizations of rationale representations.

3.3 MI Estimation Quality

In this section, we first evaluate CLUB_NCE on simulation studies with two toy tasks. Then, we show the performance of CLUB_NCE and other MI minimization methods on two real-world applications which described previously.

3.3.1 Simulated Studies

Here, we introduce the experimental simulated datasets and comparison methods. First, following the setup from Poole et al. [28] and Cheng et al. [10], we experiment a dataset where x and y are sampled from a correlated Gaussian distribution: $x \sim \mathcal{N}(0, I_d)$, $y \sim \mathcal{N}\left(\rho x, \left(1-\rho^2\right) I_d\right)$, where ρ is the correlation and d is the dimensionality. In the first toy task (denoted by **Gaussian**), we sample each dimension of $(x; y)$ from this simulated dataset, and the second task (**Cubic**) is

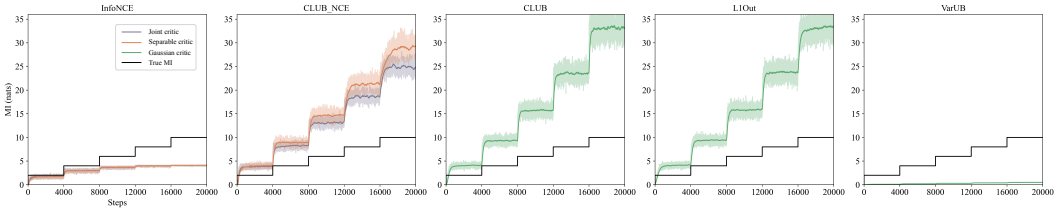

Figure 6: Performance of MI estimation approaches on the **Gaussian** task. Among them, the true MI values are the black line. The estimated values are drawn as light shadow curves while the dark shadow ones represent the local averages of estimated values, with a bandwidth 200.

the same as the first while applying the transformation $y \to y^3$ (i.e., sample $(x; y^3)$ from the same Gaussian distribution). Meanwhile, the value of this mutual information is invariant (i.e., $I(x, y) = I(x, y^3)$) [18]. Under Gaussian distributions, the ground truth of MI is tractable and can be calculated as $I(x, y) = -\frac{d}{2} \log \left(1 - \rho^2\right)$ [28]. For both of the tasks, we set the dimension to $d = 20$, hidden size to 128 and batch size to 64. Besides, we set the initial MI value as 2 and increase it by 2 every 4k iterations, for a total of 20k iterations.

Then, we present the comparison methods including InfoNCE [26], CLUB [10], VarUB [1], and L1Out [28]. Among them, CLUB and InfoNCE have been described in section 2.3.1, and VarUB ($I_{varub} = \mathbb{E}_{p(x,y)} \left[\log \frac{p(y|x)}{r(y)}\right]$) and L1Out ($I_{l1out} = \frac{1}{N} \sum_{i=1}^{N} \left[\log \frac{p(y_i|x_i)}{\frac{1}{N-1} \sum_{j \neq i} p(y_i|x_j)}\right]$) are both MI minimization methods which are adopted to estimate the upper bound of MI, where $p(y|x)$ is commonly parameterized by a Gaussian family [10] and $r(y)$ is fixed as a standard normal distribution.

3.3.2 Experimental Results on Simulated Studies

In this section, we mainly report the experimental analysis on the **Gaussian** task, the analysis on **Cubic** is similar and can be found in Appendix A.3. Figure 6 shows the performance of our CLUB_NCE and other MI estimation approaches. Among them, the joint critic is a single Multilayer Perceptron (MLP) and takes $[x; y]$ as the input, where " ; " represents the concatenate operation, and the separable critic is $f(x, y) = g_1(x)^T g_2(y)$, where g_1 and g_1 are two different MLPs. The gaussian critic is parameterized by a Gaussian family (i.e., $\mathcal{N}\left(y \mid \mu(x), \sigma^2(x) \cdot I\right)$), where the mean and variance are calculated by neural networks. We can observe that the InfoNCE values are under the true MI while CLUB_NCE are above it, which illustrates CLUB_NCE is an upper bound of MI. The estimation of VarUB has low value, indicating this method could not remain an upper bound in some cases. The CLUB and L1Out has high bias and variance when the ground truth MI is high, whereas CLUB_NCE has the lower ones, especially the variance. It might have benefited from the strength of InfoNCE which has a low variance.

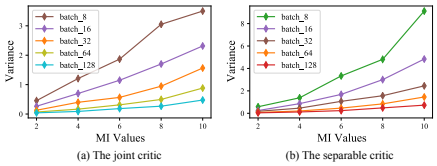

Figure 7: Variance of the CLUB_NCE on the **Gaussian** with different batch sizes in the range {8, 16, 32, 64, 128}.

To further validate the influence of InfoNCE on CLUB_NCE, we first analyze the relationship between them theoretically, and then conduct a comparison among the CLUB_NCE with different batch sizes. Specifically, we review the InfoNCE shown in Eq (3):

$$I_{nce} = \frac{1}{N} \sum_{i=1}^{N} \log \frac{e^{f(x_i, y_i)}}{\frac{1}{N} \sum_{j=1}^{N} e^{f(x_i, y_j)}} = \frac{1}{N} \sum_{i=1}^{N} \log \frac{e^{f(x_i, y_i)}}{\sum_{j=1}^{N} e^{f(x_i, y_j)}} + \log N, \tag{7}$$

where the first term is less than 0 and I_{nce} is upper bounded by $\log N$ [26]. Therefore, the performance of InfoNCE is influenced by batch sizes, where the larger the batch size N, the lower variance between the estimated MI and the true one [28]. In other words, since CLUB_NCE adopts the trained critic by InfoNCE (Algorithm 1), the value of I_{club} should have lower variance as the batch size increases. Therefore, we conduct an experiment to estimate CLUB_NCE by adopting various batch sizes. As shown in Figure 7, we calculate the variance between the estimated MI values I_{club} and the theoretical MI ones on the **Gaussian** task with different batch sizes. From the observation, we can conclude that CLUB_NCE with larger batch size achieves the lower variance, which supports the previous theoretical analysis.

3.3.3 Real-world Applications

We compare CLUB_NCE with CLUB, L1Out and VarUB on beer reviews prediction and legal judgment prediction. Specifically, we replace CLUB_NCE in DARE with other MI minimization methods, and denote them as DARE (CLUB), DARE (L1Out) and DARE (VarUB). The experimental results are reported in Table 1 and Table 2. Among them, as DARE (VarUB) fails to converge within epochs in the experimental setup, we do not present its result in the table. From the observation, DARE performs the best which indicates our CLUB_NCE is more effective than others methods at minimizing MI on real-world applications. Besides, these variants of DARE fail to improve on some tasks (e.g., DARE (L1Out) in Palate prediction and Terms of Penalty prediction), and are even less effective than other *selector-predictor* models, illustrating it is crucial for DARE to choose a superior MI estimator to effectively minimize the MI.

4 Related Work

Rationale Extraction. Rationale extraction methods can provide the model explainability by extracting important features of inputs and many researchers [20, 3, 36, 7, 8, 32, 37] have been attracted. Among them, Lei et al. [20] proposed a classical framework for rationales extraction with a selector and predictor. Following this framework, Bastings et al. [3] introduced a HardKuma distribution for reparameterized gradient estimates and facilitated end-to-end differentiability of this framework. Paranjape et al. [27] and Chen et al. [9] managed trade-off between extracting sparse rationales and generating accurate results by the information bottleneck theory. Since this type of framework failed to exploit the information of the original text, several guider methods [31, 6, 16] have been proposed, which adopted the whole text to guide the rationale extraction. Therein, Sha et al. [31] introduced an adversarial-based technique to make the selector-predictor model learn from the guider model. Cao et al. [6] minimized the KL divergence between the two types models to reduce their difference. Although extensive research has been carried out on the rationale extraction, few considered the information of non-rationale tokens containing rich information for extracting rationales.

Disentangled Representation Learning. Disentangled representation learning (DRL) [5, 15, 13, 2] maps different aspects of data into independent latent vectors by adding constraints on the embedding space and has attracted much attention. Recently, information theory has been widely adopted in the applications of DRL. Among them, Sanchez et al. [30], Cheng et al. [11] and Huang et al. [17] maximized MI between the original input and its different aspects to retain as much as possible information of the input. And Cheng et al.[10] and Colombo et al. [12] reduced the dependency among the different aspects of the input to make them independent by minimizing the MI.

Mutual Information. Mutual information (MI) is adopted to measure the dependency between random variables. Since computing the values of MI is challenging, and many researches utilized the MI as a regularizer in loss functions, and not concerned with its precise value, recent works focused on MI maximization and minimization. Nguyen et al. [25] studied a lower bound of MI based on the MI f-divergence representation for maximizing MI. Belghazi et al. [4] proposed a mutual information estimator (MINE) which estimated the lower bound of MI. Similarly, InfoNCE was introduced by Oord et al. [26], which adopted the negative sampling estimation to maximize MI. For MI minimization, Poole et al. [28] studied an upper bound of MI estimator L1Out based on reviewing the existing variational bounds on MI. And a contrastive log-ratio upper bound (CLUB) was proposed by [11, 10] to minimize MI. Besides, based on the KL divergence and Renyi divergence, Colombo et al. [12] derived a MI estimator on the basis of a variational upper bound.

5 Conclusion

In this paper, we proposed a disentanglement-augmented rationale extraction method (DARE) which squeezed more information from the original input by exploiting the non-rationale tokens. To be specific, DARE disentangled the textual input into two parts (i.e., the rationale representations and the non-rationale ones), and minimized the mutual information (MI) between the two disentanglements for extracting more comprehensive rationales. Besides, to improve the performance of minimizing MI, we introduced a new MI minimization method CLUB_NCE based on exploring the relationship between InfoNCE and CLUB. Experimental results on two real-world datasets and a simulated dataset demonstrated the effectiveness of our proposed method.

References

[1] Alexander A. Alemi, Ian Fischer, Joshua V. Dillon, and Kevin Murphy. Deep variational information bottleneck. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*, 2017.

[2] Cher Bass, Mariana da Silva, Carole Sudre, Petru-Daniel Tudosiu, Stephen Smith, and Emma Robinson. Icam: Interpretable classification via disentangled representations and feature attribution mapping. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

[3] Jasmijn Bastings, Wilker Aziz, and Ivan Titov. Interpretable neural predictions with differentiable binary variables. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (ACL)*, July 2019.

[4] Mohamed Ishmael Belghazi, Aristide Baratin, Sai Rajeshwar, Sherjil Ozair, Yoshua Bengio, Aaron Courville, and Devon Hjelm. Mutual information neural estimation. In *International Conference on Machine Learning*, pages 531–540. PMLR, 2018.

[5] Yoshua Bengio, Aaron Courville, and Pascal Vincent. Representation learning: A review and new perspectives. *IEEE transactions on pattern analysis and machine intelligence*, 35(8): 1798–1828, 2013.

[6] Nicola De Cao, Michael Sejr Schlichtkrull, Wilker Aziz, and Ivan Titov. How do decisions emerge across layers in neural models? interpretation with differentiable masking. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, 2020.

[7] Shiyu Chang, Yang Zhang, Mo Yu, and Tommi S. Jaakkola. A game theoretic approach to class-wise selective rationalization. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, 2019.

[8] Shiyu Chang, Yang Zhang, Mo Yu, and Tommi S. Jaakkola. Invariant rationalization. In *Proceedings of the 37th International Conference on Machine Learning, (ICML)*, 2020.

[9] Hanjie Chen and Yangfeng Ji. Learning variational word masks to improve the interpretability of neural text classifiers. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 4236–4251, 2020.

[10] Pengyu Cheng, Weituo Hao, Shuyang Dai, Jiachang Liu, Zhe Gan, and Lawrence Carin. Club: A contrastive log-ratio upper bound of mutual information. In *International Conference on Machine Learning (ICML)*, pages 1779–1788. PMLR, 2020.

[11] Pengyu Cheng, Martin Renqiang Min, Dinghan Shen, Christopher Malon, Yizhe Zhang, Yitong Li, and Lawrence Carin. Improving disentangled text representation learning with information-theoretic guidance. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL)*, 2020.

[12] Pierre Colombo, Pablo Piantanida, and Chloé Clavel. A novel estimator of mutual information for learning to disentangle textual representations. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, ACL/IJCNLP 2021, Virtual Event, August 1-6, 2021*, 2021.

[13] Shuyang Gao, Rob Brekelmans, Greg Ver Steeg, and Aram Galstyan. Auto-encoding total correlation explanation. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 1157–1166. PMLR, 2019.

[14] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. *Advances in neural information processing systems*, 27, 2014.

[15] Irina Higgins, David Amos, David Pfau, Sébastien Racanière, Loïc Matthey, Danilo J. Rezende, and Alexander Lerchner. Towards a definition of disentangled representations. *CoRR*, 2018.

[16] Yongfeng Huang, Yujun Chen, Yulun Du, and Zhilin Yang. Distribution matching for rationalization. In *Thirty-Fifth AAAI Conference on Artificial Intelligence, (AAAI)*, 2021.

[17] Zhenya Huang, Xin Lin, Hao Wang, Qi Liu, Enhong Chen, Jianhui Ma, Yu Su, and Wei Tong. Disenqnet: Disentangled representation learning for educational questions. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, pages 696–704, 2021.

[18] Alexander Kraskov, Harald Stögbauer, and Peter Grassberger. Estimating mutual information. *Physical review E*, 69(6):066138, 2004.

[19] Ponnambalam Kumaraswamy. A generalized probability density function for double-bounded random processes. *Journal of hydrology*, 46(1-2):79–88, 1980.

[20] Tao Lei, Regina Barzilay, and Tommi Jaakkola. Rationalizing neural predictions. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2016.

[21] Yi-Shan Lin, Wen-Chuan Lee, and Z Berkay Celik. What do you see? evaluation of explainable artificial intelligence (xai) interpretability through neural backdoors. *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, 2021.

[22] Bingfeng Luo, Yansong Feng, Jianbo Xu, Xiang Zhang, and Dongyan Zhao. Learning to predict charges for criminal cases with legal basis. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2017.

[23] Siwen Luo, Hamish Ivison, Caren Han, and Josiah Poon. Local interpretations for explainable natural language processing: A survey. *arXiv preprint arXiv:2103.11072*, 2021.

[24] Julian J. McAuley, Jure Leskovec, and Dan Jurafsky. Learning attitudes and attributes from multi-aspect reviews. In *12th IEEE International Conference on Data Mining, ICDM 2012, Brussels, Belgium, December 10-13, 2012*.

[25] XuanLong Nguyen, Martin J Wainwright, and Michael I Jordan. Estimating divergence functionals and the likelihood ratio by convex risk minimization. *IEEE Transactions on Information Theory*, 56(11):5847–5861, 2010.

[26] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

[27] Bhargavi Paranjape, Mandar Joshi, John Thickstun, Hannaneh Hajishirzi, and Luke Zettlemoyer. An information bottleneck approach for controlling conciseness in rationale extraction. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, November 2020.

[28] Ben Poole, Sherjil Ozair, Aaron Van Den Oord, Alex Alemi, and George Tucker. On variational bounds of mutual information. In *International Conference on Machine Learning (ICML)*, pages 5171–5180. PMLR, 2019.

[29] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. " why should i trust you?" explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, 2016.

[30] Eduardo Hugo Sanchez, Mathieu Serrurier, and Mathias Ortner. Learning disentangled representations via mutual information estimation. In *Computer Vision - ECCV 2020 - 16th European Conference, Glasgow, UK, August 23-28, 2020, Proceedings, Part XXII*, 2020.

[31] Lei Sha, Oana-Maria Camburu, and Thomas Lukasiewicz. Learning from the best: Rationalizing prediction by adversarial information calibration. In *Proceedings of the 35th AAAI Conference on Artificial Intelligence (AAAI)*, 2021.

[32] Marcos V. Treviso and André F. T. Martins. The explanation game: Towards prediction explainability through sparse communication. In *Proceedings of the Third BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP*, 2020.

[33] Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(11), 2008.

[34] Chaojun Xiao, Haoxi Zhong, Zhipeng Guo, Cunchao Tu, Zhiyuan Liu, Maosong Sun, Yansong Feng, Xianpei Han, Zhen Hu, Heng Wang, et al. Cail2018: A large-scale legal dataset for judgment prediction. In *arXiv preprint arXiv:1807.02478*, 2018.

[35] Nuo Xu, Pinghui Wang, Long Chen, Li Pan, Xiaoyan Wang, and Junzhou Zhao. Distinguish confusing law articles for legal judgment prediction. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3086–3095. Association for Computational Linguistics, 2020.

[36] Mo Yu, Shiyu Chang, Yang Zhang, and Tommi S Jaakkola. Rethinking cooperative rationalization: Introspective extraction and complement control. In *Empirical Methods in Natural Language Processing (EMNLP)*, 2019.

[37] Mo Yu, Yang Zhang, Shiyu Chang, and Tommi Jaakkola. Understanding interlocking dynamics of cooperative rationalization. *Advances in Neural Information Processing Systems (NeurIPS)*, 34, 2021.

[38] Linan Yue, Qi Liu, Binbin Jin, Han Wu, Kai Zhang, Yanqing An, Mingyue Cheng, Biao Yin, and Dayong Wu. Neurjudge: A circumstance-aware neural framework for legal judgment prediction. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2021.

[39] Haoxi Zhong, Zhipeng Guo, Cunchao Tu, Chaojun Xiao, Zhiyuan Liu, and Maosong Sun. Legal judgment prediction via topological learning. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2018.

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes] We discuss the limitations in Appendix A.4.
   (c) Did you discuss any potential negative societal impacts of your work? [Yes] We discuss the broader impacts in Appendix A.4.
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [Yes] Please see Section 2.
   (b) Did you include complete proofs of all theoretical results? [Yes] Please see Section 2.
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes] Please find the code in the supplementary material.
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] Please see Section 3.1.1, 3.2.1 and Appendix A.1.
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [Yes] Please see Table 1 and Table 2.
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes] We run all experiments on a single V100 GPU.
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes]

(b) Did you mention the license of the assets? [Yes] Although the original BeerAdvocate has been removed by the authors of this dataset, we can obtain BeerAdvocate from `http://people.csail.mit.edu/taolei/beer/`, which is published by [20]. CAIL2018 has been published in `https://github.com/china-ai-law-challenge/CAIL2018`.

(c) Did you include any new assets either in the supplemental material or as a URL? [N/A]

(d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [Yes] Please see Section 3.1, Section 3.2.

(e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [Yes] To our best knowledge, their exists no personally identifiable information in both BeerAdvocate and CAIL.

5. If you used crowdsourcing or conducted research with human subjects...

(a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]

(b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]

(c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]