MARE: Multi-Aspect Rationale Extractor on Unsupervised Rationale Extraction

Han Jiang and Junwen Duan* and Zhe Qu and Jianxin Wang
Hunan Provincial Key Lab on Bioinformatics, School of Computer Science and Engineering,
Central South University, Changsha, Hunan, China
{jh-better, jwduan, zhe_qu}@csu.edu.cn, jxwang@mail.csu.edu.cn

Abstract

Unsupervised rationale extraction aims to extract text snippets to support model predictions without explicit rationale annotation. Researchers have made many efforts to solve this task. Previous works often encode each aspect independently, which may limit their ability to capture meaningful internal correlations between aspects. While there has been significant work on mitigating spurious correlations, our approach focuses on leveraging the beneficial internal correlations to improve multi-aspect rationale extraction. In this paper, we propose a Multi-Aspect Rationale Extractor (MARE) to explain and predict multiple aspects simultaneously. Concretely, we propose a Multi-Aspect Multi-Head Attention (MAMHA) mechanism based on *hard deletion* to encode multiple text chunks simultaneously. Furthermore, multiple special tokens are prepended in front of the text with each corresponding to one certain aspect. Finally, multi-task training is deployed to reduce the training overhead. Experimental results on two unsupervised rationale extraction benchmarks show that MARE achieves state-of-the-art performance. Ablation studies further demonstrate the effectiveness of our method. Our codes have been available at https://github.com/CSU-NLP-Group/MARE.

1 Introduction

Deep learning text classification systems have achieved remarkable performance in recent years (Kim, 2014; Devlin et al., 2019). However, their black-box nature has been widely criticized. Finding a sufficient approach to open the black box is urgent and significant.

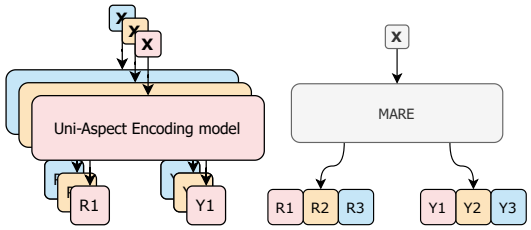
Unsupervised rationale extraction (Lei et al., 2016) is an explanation approach that aims to extract text snippets from input text to support model predictions without explicit rationale annotation. Previous researchers (Liu et al., 2022; Jiang et al., 2023) have made many efforts to improve the rationalization performance of their models. However, as shown in Figure 1a, existing rationale extraction models are uni-aspect encoding models, which can only predict and interpret one aspect of the text at a time. In real-world scenarios, one text often contains multiple aspects of an object. Table 1 shows an example from the BeerAdvocate dataset (McAuley et al., 2012), where blue, red, and cyan represent the aspects of *Appearance*, *Aroma*, and *Palate*, respectively. The highlighted segments in the text are the rationales corresponding to each aspect. For instance, "*pours a murky orangish-brown color with a white head* ." explains why the label for *Appearance* is Positive. In this case, traditional uni-aspect rationale extraction models would require three independently trained models to predict and interpret all three aspects, which is labor-intensive and time-consuming and limits their downstream applications. Furthermore, uni-aspect models encode each aspect independently ignoring their internal correlation.

Example
Appearance: Positive
Aroma: Positive
Palate: Positive
Text: thanks to bman1113vr for sharing this bottle . pours a murky orangish-brown color with a white head . the aroma is tart lemons . the flavor is tart lemons with some oak-aged character . the beer finishes very dry . medium mouthfeel and medium carbonation .

Table 1: A multi-aspect example from the BeerAdvocate dataset (McAuley et al., 2012). Blue, red, and cyan represent the aspects of *Appearance*, *Aroma*, and *Palate*, respectively.

To address these problems, we propose the Multi-Aspect Rationale Extractor (**MARE**). As shown in Figure 1b, MARE can encode all aspects simultaneously by prepending multiple special tokens

Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing, pages 11734–11745
November 12-16, 2024



(a) Typical uni-aspect encoding models (b) MARE (*ours*)

Figure 1: Comparison of our methods (MARE) with previous typical uni-aspect encoding models.

to the input text, each corresponding to a specific aspect. This approach enables multi-aspect encoding in one model. Furthermore, MARE introduces a Multi-Aspect Multi-Head Attention (**MAMHA**) mechanism for collaborative encoding across aspects. This mechanism allows the model to capture interactions and dependencies between different aspects, leading to more accurate predictions and rationales. Finally, inspired by multi-task learning, MARE iteratively accesses training data for different aspects, reducing the overall training cost.

We validate the effectiveness of MARE on two unsupervised rationale extraction benchmarks: BeerAdvocate (McAuley et al., 2012) and Hotel Review (Wang et al., 2010). Results show that MARE outperforms existing state-of-the-art methods across multiple evaluation metrics. Ablation studies further demonstrate the effectiveness of MARE. Our main contributions are as follows:

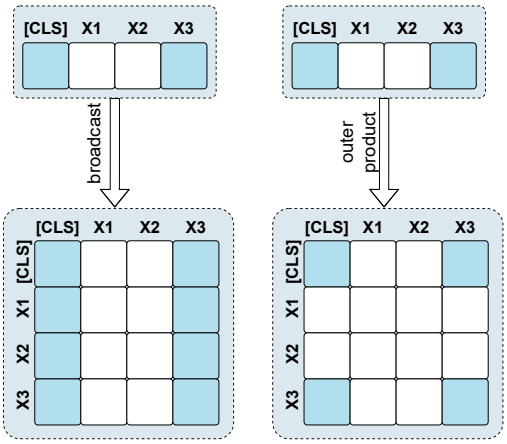
- We introduce MARE, a Multi-Aspect Rationale Extractor that generates predictions and rationales for multiple aspects simultaneously.
- We deploy the multi-task training to reduce the training cost and expand the model applicability. Compared to multi-aspect collaborative training, it saves 17.9% and 25.2% of memory usage and training time, respectively.
- Extensive experiments on BeerAdvocate and Hotel Review datasets demonstrate MARE's superiority, with a notable 5.4% improvement in token-level F1 score. Ablation studies further validate the effectiveness of each component in MARE.

2 Related Work

The rationalization framework, known as RNP (Lei et al., 2016), assumes that any unselected input has no contribution to the prediction and achieves remarkable performance on this task. However, RNP still has many weaknesses. Various approaches have been proposed to improve RNP In different dimensions.

Gradient Flows The RNP framework utilizes REINFORCE (Williams, 1992) to overcome the non-differentiable problem, but this leads to training instability and poor performance. Hard-Kuma (Bastings et al., 2019) introduces reparameterization tricks and replaces the Bernoulli distribution with the rectified Kumaraswamy distribution, which stabilizes the training process. In FR (Liu et al., 2022), the encoder's parameter is shared between the generator and predictor. This ensures that the encoder's gradient is more reasonable because it can see both full texts and rationales. 3Players (Yu et al., 2019) forces the complementary rationale to be meaningless, resulting in more meaningful generated rationales. Our research is orthogonal with these methods.

Interlocking The interlocking problem was initially proposed by A2R (Yu et al., 2021). This problem arises when the generator fails to identify important tokens, leading to sub-optimal rationales and consequently affecting the performance. Many researchers have developed approaches to address this issue (Huang et al., 2021; Yu et al., 2021; Liu et al., 2023a). DMR (Huang et al., 2021) aimed to align the distributions of rationales with the full input text in the output space and feature space. A2R (Yu et al., 2021) enhances the predictor's understanding of the full text by introducing a soft rationale. MGR (Liu et al., 2023a) involves multiple generators with different initializations to allow the predictor to see various rationales, alleviating the interlocking problem. DR (Liu et al., 2023b) limits the Lipschitz constant of the predictor, making the whole system more robust. DAR (Liu et al., 2024a) deploys a pre-trained discriminator to align the selected rationale and the original input. MCD (Liu et al., 2024b) proposes the minimum conditional dependence criterion to overcome the issues of the maximum mutual information (MMI) criterion. YOFO (Jiang et al., 2023) eliminates interlocking by simultaneously predicting and interpreting. YOFO deploys pre-trained language models as its backbone and uses token deletion strategies between layers to erase unimportant tokens. the remaining tokens in the final layer are seen as rationales.



(a) Attention Mask Deletion (b) Hard Deletion

Figure 2: Attention mask visualization. left: attention mask in Attention Mask Deletion. right: attention mask in Hard Deletion.

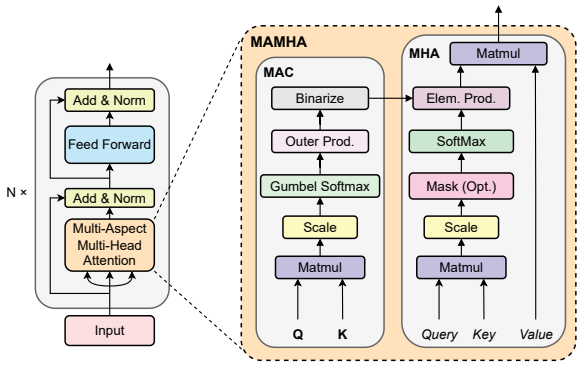


Figure 3: Overall model architecture. left: the overall model architecture of MARE. right: the computational graph of MAMHA.

In the domain of multi-aspect rationale extraction, several approaches have been proposed. MTM (Antognini et al., 2021) introduced a method for a multi-aspect explanation of target variables from documents, which bears some similarities to our work. Their approach, like ours, aims to provide explanations for multiple aspects simultaneously. However, there are key differences in the model architecture and methodology. 1. Model architecture: Unlike two-stage models that generate rationales and labels sequentially, MARE is a single-stage model that generates both simultaneously. 2. Base model: While some existing approaches use LSTM or CNN architectures, MARE leverages the power of pre-trained transformer models like BERT. 3. Aspect assignment: Our method allows for a token to be assigned to multiple aspects independently, whereas some existing methods normalize probabilities across aspects, limiting each token to a single aspect.

This paper focuses on the efficiency of the multi-aspect scenarios. All the above models are uni-aspect encoding models, where one model can only encode one aspect of data. MARE is a multi-aspect collaborative encoding model designed to encode multiple aspects of data simultaneously.

3 Problem Definition

Existing uni-aspect encoding models extract rationales $\mathbf{z}_i$ from the input $\mathbf{x}$ and predict the label $\mathbf{y}_i$ for the i-th aspect. Formally, they can be expressed as $P(\mathbf{y}_i, \mathbf{z}_i \mid \mathbf{x}; \theta_i)$, where θ_i represents the parameters of the model for the i-th aspect. To obtain the rationales and predictions for all k aspects, k independently trained models are required: $\{P(\mathbf{y}_1, \mathbf{z}_1 \mid \mathbf{x}; \theta_1), \cdots, P(\mathbf{y}_k, \mathbf{z}_k \mid \mathbf{x}; \theta_k)\}$. However, this approach is time-consuming and computationally expensive.

To address this issue, we propose a multi-aspect rationale extraction task, where the rationales and predictions for all aspects can be generated simultaneously. This can be formalized as $P(\mathbf{y}_1, \mathbf{z}_1, \cdots, \mathbf{y}_k, \mathbf{z}_k \mid \mathbf{x}; \theta)$, where θ represents the parameters of the multi-aspect rationale extraction model. By utilizing a single model to extract rationales and make predictions for all aspects concurrently, we aim to improve the efficiency and reduce computational costs compared.

4 Method

This paper proposes a Multi-Aspect Rationale Extractor (MARE), which can simultaneously predict and interpret multiple aspects of text. As shown in the left part of Figure 3, MARE is based on an encoder-based pre-trained language model and achieves multi-aspect collaborative encoding through a Multi-Aspect Multi-Head Attention (MAMHA) mechanism. Additionally, MARE employs multi-task training during the training process, significantly reducing the training cost.

4.1 Hard Deletion for Complete Token Removal

Selecting rationales without explicit annotations can be challenging. We follow the previous work (Jiang et al., 2023) where unimportant tokens are gradually erased. However, directly multiplying hidden states by the token mask harms rationalization performance (Jiang et al., 2023). Attention

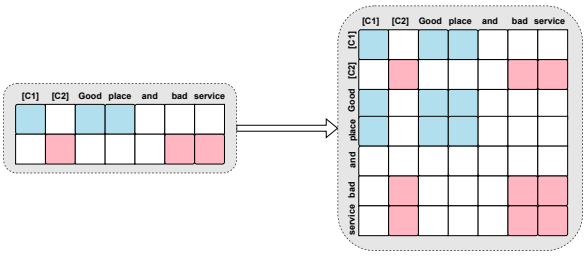


Figure 4: A example for Multi-Aspect Controller. left: The token mask for each aspect. "Good place" and "bad service" stands for the rationales of *location* and *service* aspect, respectively. right: The attention mask is obtained by performing an outer product operation on token masks.

Mask Deletion (AMD) (Jiang et al., 2023) avoids this problem by setting attention scores of masked tokens to 0. Concretely, assuming $\mathbf{m}_i \in [0,1]^L$ represents the token mask in the i-th layer and $\mathbf{A}_i^j \in \mathbb{R}^{L\times L}$ is the attention score matrix of the j-th head in the i-th layer, the final attention score matrix is $\tilde{\mathbf{A}}_i^j = \mathbf{A}_i^j \cdot \mathbf{m}_i \in \mathbb{R}^{L\times L}$. Through AMD, remaining tokens interact while deleted ones are invisible.

However, AMD suffers from an "incomplete deletion" problem, where deleted tokens can still be partially represented by remaining ones due to the broadcast operation. As shown in Figure 2a, although "X1" and "X2" are masked, they can still be indirectly represented by the weighted sum of "[CLS]" and "X3". Although this allows the model to retain more information, it hinders multi-aspect collaborative encoding.

To address this issue, we propose **Hard Deletion**, which uses an outer product operation to completely erase deleted tokens (Figure 2b). "X1" and "X2" are represented by all-zero vectors, ensuring complete removal.

4.2 Multi-Aspect Multi-Head Attention

Inspired by hard deletion, we propose the multi-aspect multi-head attention (MAMHA) mechanism to encode multiple text segments simultaneously. As shown in the right part of Figure 3, MAMHA consists of a Multi-Aspect Controller (MAC) and the traditional multi-head attention (MHA) mechanism.

4.2.1 Multi-Aspect Controller (MAC)

MAC assists MHA in separately encoding different text segments by generating aspect-specific attention masks based on token masks for each aspect. This allows tokens within the same aspect to interact while isolating tokens from different aspects, enabling MHA to achieve multi-aspect collaborative encoding.

Figure 4 illustrates an example where "good place" and "bad service" are rationales for the "*location*" and "*service*" aspects, respectively. The final attention mask, obtained through an outer product operation, creates two separate segments. Words within each segment interact, while words from different segments remain independent. Special classification tokens "[C1]" and "[C2]" collect information from their respective aspects, allowing MHA to encode two aspects simultaneously.

This method can be extended to k aspects by dividing the text into k segments and appending k special tokens. Note that if MAC employs AMD, tokens from different aspects cannot be fully isolated, leading to confusion and hindering multi-aspect collaborative encoding (further discussed in Section 6.2.2).

4.2.2 Computation Process of MAC

The computation process of MAC is shown in the right part of Figure 3. Assuming $\mathbf{H}_i$ represents the hidden states of the i-th layer, its first k vectors $\{\mathbf{h}_i^0, \cdots, \mathbf{h}_i^{k-1}\}$ are representations of special tokens. For the j-th aspect, mapping functions g_{query}^j and g_{key}^j map special and normal tokens to $\mathbf{Q}$ and $\mathbf{K}$, respectively. The similarity between special and normal tokens is calculated, and the gumbel-softmax technique determines the token's aspect assignment (Equations (1)-(4)).

$$\mathbf{Q} = \{g_{query}^0(\mathbf{h}_i^0), \cdots, g_{query}^k(\mathbf{h}_i^{k-1})\} \quad (1)$$

$$\mathbf{K} = \{g_{key}^0(\mathbf{H}_i[k:]), \cdots, g_{key}^k(\mathbf{H}_i[k:])\} \quad (2)$$

$$\mathbf{scores} = \frac{\mathbf{Q}\cdot\mathbf{K}^\mathsf{T}}{\sqrt{d}} \quad (3)$$

$$\mathbf{m} = \text{gumbel\_softmax}(\mathbf{scores}, dim = -1) \quad (4)$$

, where d and L mean the vector's dimension and the text's length, respectively. $[\cdot]$ represents slicing operation, $\mathbf{m} \in \{0,1\}^{k\times L}$ stands for the token mask, and $m[i,j] = 1$ indicates that the j-th token is selected as the rationale of the i-th aspect. The mask $\mathbf{m}$ evolves during training, starting as a near-full 1 vector and gradually becoming more selective. The Gumbel-Softmax output is binarized to produce the final mask.

MAC adopts the outer product operation to match the shape of the attention score matrix in MHA (Equation 5). When $M'[i,j] \neq 0$, the token is selected as a rationale in at least one aspect and

should not be deleted (Equation (6)). The binarization operation in Equation (6) is non-differentiable, so straight-through is used for gradient estimation. Finally, the mask is multiplied by the attention score matrix to perform token deletion (Equations (7)-(9)).

$$\mathbf{M}' = \mathbf{m}^{\mathsf{T}} \cdot \mathbf{m} \in \mathbb{Z} \cap [0, k]^{L \times L} \tag{5}$$

$$\tilde{M}[i,j] = \begin{cases} 0, \text{ If } M'[i,j] = 0 \\ 1, \text{ Otherwise} \end{cases} \tag{6}$$

$$\mathbf{M} = \tilde{\mathbf{M}} + \mathbf{M}' - \text{StopGrad}(\mathbf{M}') \in [0,1]^{L \times L} \tag{7}$$

$$\tilde{\mathbf{A}}_i^h = \mathbf{A}_i^h \odot \mathbf{M}, \text{ for } h \text{ in } 1, 2, ..., H \tag{8}$$

$$\mathbf{H}_i = \text{PLM}_i(\mathbf{H}_{i-1}; \tilde{\mathbf{A}}_i), \text{ for } i \text{ in } 1, 2, ..., N \tag{9}$$

, where StopGrad(X) represents stopping the X's gradient calculation. $\mathbf{A}_i^h$ and $\tilde{\mathbf{A}}_i^h$ represent the initial and final attention score matrices of the h-th attention header in the i-th layer, respectively. $\mathbf{H}_i$ represents the hidden layer representation of the i-th layer.

4.3 Multi-Task Training

Using labels from various aspects simultaneously during training may not be feasible, as datasets like Hotel Review (Wang et al., 2010) only have annotations for one aspect per sample. Multi-task training allows MARE to focus on the aspect corresponding to the current batch, avoiding the need to encode aspects with missing labels. If the batch comes from the j-th aspect, only the corresponding mapping functions g_q^j and g_k^j are used (Equations (10)-(11)).

$$\mathbf{Q} = g_{query}^j(\mathbf{H}_i[j-1:j]) \tag{10}$$

$$\mathbf{K} = g_{key}^j(\mathbf{H}_i[k:]) \tag{11}$$

At inference time, we do not explicitly control the sparsity level. Instead, the trained mapping functions g_q and g_k directly select tokens they identify as explanations. This means that the proportion of selected tokens for each aspect in a single sample is not strictly fixed. It can vary based on the content, and may even be 0% if the model determines there is no relevant description for a particular aspect.

4.4 Overall Loss

Our loss function consists of three components: cross-entropy loss (L_{CE}), sparsity penalty (L_{sparse}), and contiguous penalty (L_{cont}). The full loss function is:

$$L = L_{CE} + \beta L_{sparse} + \gamma L_{cont} \tag{12}$$

$$L_{CE} = \frac{1}{C} \sum_{i=1}^{C} y_i \log p_i \tag{13}$$

$$L_{sparse} = \frac{1}{N} \sum_{i=1}^{N} |\frac{1}{L} \sum_{j=1}^{L} m_i^j - l_i| \tag{14}$$

$$L_{cont} = \frac{\sum_{i=1}^{N} \sum_{j=1}^{L} |m_i^{j+1} - m_i^j|}{N(L-1)} \tag{15}$$

, where s is a predefined sparsity level, β and γ are hyperparameters that balance these terms. We employ a Cliff decay strategy as illustrated in Appendix B, where token deletion begins after a specified layer in the network.

5 Experiments

5.1 Experimental Setup

Datasets We performed experiments on two commonly used unsupervised rationale extraction datasets: BeerAdvocate (McAuley et al., 2012) and the Hotel Review dataset (Wang et al., 2010).

The BeerAdvocate dataset (McAuley et al., 2012) is a multi-aspect sentiment prediction dataset. It consists of texts along with corresponding aspect scores ranging from 0 to 1, including aspects such as *appearance*, *aroma*, and *palate*. The training and validation sets do not have labeled rationales, but the test set contains 994 samples with rationale annotations for all aspects. Notably, the scores across different aspects within the same sample exhibit high correlation, resulting in highly spurious correlations. For the BeerAdvocate dataset, we conducted experiments on the decorrelated version proposed by Lei et al.. We binarized the dataset into binary classification tasks using a positive threshold of 0.6 and a negative threshold of 0.4 (Bao et al., 2018). We run our model, MARE, on two sparsity levels: high-sparse and low-sparse. In the high-sparse decorrelated dataset, the sparsity level approximates the sparsity for golden rationales in the test set. In the low-sparse decorrelated dataset, the sparsity level is comparatively lower but allows for convenient comparisons with previous works.

The Hotel Review dataset (Wang et al., 2010) is another widely used dataset for multi-aspect sentiment classification and rationale extraction. It includes texts along with three aspect labels: *location*, *service*, and *cleanliness*. In addition to the

Methods	Appearance					Aroma					Palate					Avg
	S	ACC	P	R	F1	S	ACC	P	R	F1	S	ACC	P	R	F1	F1
RNP(Lei et al., 2016)	18.7	84.0	72.0	72.7	72.3	15.1	85.2	59.0	57.2	58.1	13.4	90.0	63.1	68.2	65.5	65.6
DMR(Huang et al., 2021)	18.2	-	71.1	70.2	70.7	15.4	-	59.8	58.9	59.3	11.9	-	53.2	50.9	52.0	60.7
A2R(Yu et al., 2021)	18.4	83.9	72.7	72.3	72.5	15.4	86.3	63.6	62.9	63.2	12.4	81.2	57.4	57.3	57.4	64.5
FR(Liu et al., 2022)	18.4	87.2	82.9	82.6	82.8	15.0	88.6	74.7	72.1	73.4	12.1	89.7	67.8	66.2	67.0	74.4
MGR(Liu et al., 2023a)	18.4	86.1	83.9	83.5	83.7	15.6	86.6	76.6	76.5	76.5	12.4	85.1	66.6	66.6	66.6	75.6
DR(Liu et al., 2023b)	18.6	85.3	84.3	84.8	84.5	15.6	87.2	77.2	77.5	77.3	13.3	85.7	65.1	69.8	67.4	76.4
YOFO (Jiang et al., 2023)	18.1	85.6	91.3	87.1	89.2	15.4	86.8	**94.3**	87.9	91.0	13.2	88.4	79.5	79.0	79.2	86.5
MARE (*ours*)	17.3	85.6	**95.4**	**89.7**	**92.5**	15.4	86.0	93.9	**90.2**	**92.0**	12.7	88.0	**82.2**	**81.9**	**82.0**	**88.8**

Table 2: Results of different methods on the high-sparse decorrelated BeerAdvocate dataset.

Methods	Appearance					Aroma					Palate					Avg
	S	ACC	P	R	F1	S	ACC	P	R	F1	S	ACC	P	R	F1	F1
RNP(Lei et al., 2016)	11.9	-	72.0	46.1	56.2	10.7	-	70.5	48.3	57.3	10.0	-	53.1	42.8	47.5	53.7
CAR(Chang et al., 2019)	11.9	-	76.2	49.3	59.9	10.3	-	50.3	33.3	40.1	10.2	-	56.6	46.2	50.9	50.3
DMR(Huang et al., 2021)	11.7	-	83.6	52.8	64.7	11.7	-	63.1	47.6	54.3	10.7	-	55.8	48.1	51.7	56.9
FR(Liu et al., 2022)	12.7	83.9	77.6	53.3	63.2	10.8	87.6	82.9	57.9	68.2	10.0	84.5	69.3	55.8	61.8	64.4
MGR(Liu et al., 2023a)	13.2	82.6	75.2	53.5	62.6	12.3	84.7	80.8	63.7	71.2	10.8	80.1	51.6	44.7	47.9	60.6
DR(Liu et al., 2023b)	11.9	81.4	86.8	55.9	68.0	11.2	80.5	70.8	57.1	63.2	10.5	81.4	71.2	60.2	65.3	65.5
YOFO (Jiang et al., 2023)	13.1	87.0	97.1	66.9	79.2	12.1	86.3	94.1	68.9	79.5	10.9	87.8	**88.5**	72.7	79.8	79.5
MARE (*ours*)	13.8	86.3	**98.7**	**74.0**	**84.6**	12.2	85.9	**97.5**	**74.4**	**84.3**	10.9	88.2	87.4	**74.6**	**80.5**	**83.1**

Table 3: Results of different methods on the low-sparse decorrelated BeerAdvocate dataset.

aspect labels, the test set of this dataset also provides rationale annotations for all three aspects, with 200 samples. Since the original labels are on a scale of 0 to 5 stars, we utilize the binarized version proposed by Bao et al.. For the Hotel Review dataset, we only conducted a low-sparse experiment as the golden sparsity level is relatively low, at around 10%.

The statistics of the BeerAdvocate (McAuley et al., 2012) and Hotel Review dataset (Wang et al., 2010) are shown in Table 4.

Datasets		Train		Validation		Test	
		Pos	Neg	Pos	Neg	Pos	Neg
Beer	Appearance	16891	16891	6628	2103	923	13
	Aroma	15169	15169	6579	2218	848	29
	Palate	13652	13652	6740	2000	785	20
Hotel	Location	7236	7236	906	906	104	96
	Service	50742	50742	6344	6344	101	98
	Cleanliness	75049	75049	9382	9382	97	99

Table 4: Statistics of the BeerAdvocate and Hotel Review dataset.

Baselines We compared the performance of MARE with several state-of-the-art baselines. These baselines, including RNP (Lei et al., 2016), CAR (Chang et al., 2019), DMR (Huang et al., 2021), A2R (Yu et al., 2021), FR (Liu et al., 2022), MGR (Liu et al., 2023a) DR (Liu et al., 2023b), and YOFO (Jiang et al., 2023), were discussed in Section 2. The performance of these baselines are obtained from YOFO (Jiang et al., 2023). In MARE, we use BERT for our backbone and the balanced round-robin is equipped in the training stage. All of our experiments are conducted on NVIDIA Geforce RTX 3090 24GB. For more implementation details, please refer to Appendix A.1.

Metrics Following previous works (Jiang et al., 2023), we will use token-level F1 and accuracy for the rationalization and downstream performance. In our result tables, we define S as the sparsity level of selected rationales, computed using the formula $S = \frac{\#selected\ tokens}{\#tokens}$. P, R, and F1 represent precision, recall, and F1 score for rationale extraction, respectively. ACC and Val ACC denote the accuracy of the test and validation sets, respectively. The best performance is **Bolded** in the tables.

5.2 Main Results

5.2.1 Results on the BeerAdvocate Dataset

High-sparse Experimental results on the decorrelated BeerAdvocate dataset in the high-sparse scenario are shown in Table 2. MARE outperforms YOFO by 3.3%, 1.0%, and 2.8% in the *appearance*, *aroma*, and *palate* aspects, respectively. Meanwhile, MARE achieves the best average F1 scores among all models, particularly 88.8%. This is because MARE is a multi-aspect collaborative encoding model that captures internal correlations between all aspects and thus achieves the best performance.

Low-sparse Experimental results on the decorrelated BeerAdvocate dataset in the low-sparse

Methods	Location					Service					Cleanliness					Avg
	S	ACC	P	R	F1	S	ACC	P	R	F1	S	ACC	P	R	F1	F1
RNP(Lei et al., 2016)	8.8	97.5	46.2	48.2	47.1	11.0	97.5	34.2	32.9	33.5	10.5	96.0	29.1	34.6	31.6	37.4
DMR(Huang et al., 2021)	10.7	-	47.5	60.1	53.1	11.6	-	43.0	43.6	43.3	10.3	-	31.4	36.4	33.7	43.4
A2R(Yu et al., 2021)	8.5	87.5	43.1	43.2	43.1	11.4	96.5	37.3	37.2	37.2	8.9	94.5	33.2	33.3	33.3	37.9
FR(Liu et al., 2022)	9.0	93.5	55.5	58.9	57.1	11.5	94.5	44.8	44.7	44.8	11.0	96.0	34.9	43.4	38.7	46.9
MGR(Liu et al., 2023a)	9.7	97.5	52.5	60.5	56.2	11.8	96.5	45.0	46.4	45.7	10.5	96.5	37.6	44.5	40.7	47.5
DR(Liu et al., 2023b)	9.6	96.5	53.6	60.9	57.0	11.5	96.0	47.1	47.4	47.2	10.0	97.0	39.3	44.3	41.8	48.9
YOFO (Jiang et al., 2023)	9.7	98.0	55.7	60.4	58.0	11.9	99.5	58.3	**57.4**	**57.9**	10.6	100.0	**49.9**	**54.4**	**52.1**	56.0
MARE (*ours*)	9.7	98.0	**59.0**	**68.4**	**63.3**	10.8	99.5	**58.6**	55.2	56.8	10.6	100.0	46.8	54.0	50.1	**56.7**

Table 5: Results of different methods on the Hotel Review dataset.

MARE (*ours*)
Location: Positive ✔ **Service**: Positive ? **Cleanliness**: Positive ? **Text**: arrived very apprehensively to the hotel after reading the negative remarks . we were happily suprised . staff very pleasant , rooms and bathrooms spotlessly clean , although on the small side . our rooms had no natural light , but with the lights on were ok . air conditioning worked (was necessary in november !) , although noisy from the inside and outside where the vents are . however , the hotel is in the middle of nyc and the noise did n ' t bother us overmuch - the situation is much more important , and the jolly was in a perfect location for shopping and tourism . breakfasted across the road in the moonstruck deli (opens at 7 am for the jet lagged) . i would certainly go back there again !
Location: - **Service**: Positive ✔ **Cleanliness**: - **Text**: this is a very nice hotel with top - notch service and staff . you will pay for it , but if you want to avoid the touristy hotels of branson , this is a beautiful place to stay and eat .

Table 6: Case studies on the Hotel Review dataset.

scenario are shown in Table 3. MARE still achieves the best performance in all aspects, similar to the high-sparsity scenario. In the low-sparsity scenario, the performance gain obtained by MARE is greater than in high-sparsity scenarios. Specifically, MARE is 5.4%, 4.8%, and 0.7% higher than YOFO in the *appearance*, *aroma*, and *palate* aspects, respectively. Furthermore, MARE has a 3.6% average performance gain in token-level F1 compared to YOFO. This further demonstrates the effectiveness of MARE.

5.2.2 Results on the Hotel Review Dataset

Experimental results on the Hotel Review dataset are shown in Table 3. Although MARE is slightly inferior to YOFO in the *service* and *cleanliness* aspects, it is far superior to YOFO in the *location* aspect and its average token-level F1 score is higher than YOFO. Specifically, MARE is 1.1% and 2.0% lower than YOFO in the *service* and *cleanliness* aspects, respectively, while it is 5.3% higher than YOFO in the *location* aspect. Meanwhile, MARE is 0.7% higher than YOFO in the average token-level F1 score.

6 Analysis

In this section, we delve into a more comprehensive analysis of our methodology. In Section 6.1, we present a series of case studies derived from the Hotel Review dataset to exemplify the practical applications of our approach. In Section 6.2, we conduct an ablation study to substantiate the efficacy of our method by incrementally removing its constituent elements.

6.1 Case Study

This section visualizes several samples on the Hotel Review dataset as shown in Table 6. Blue, red, and cyan represent the *location*, *service*, and *cleanliness* aspects, respectively, and underline indicate the annotated rationales.

In the Hotel Review test set, each sample only has a uni-aspect annotation. As shown in the first case, only the *location* aspect has been annotated. However, in real scenarios, a review often describes multiple aspects. MARE extracted snippets not only about *location* but *service* and *cleanliness* which are not annotated. "*Staff very clean*" and "*rooms and bathrooms spotless clean*" demonstrate that the *service* and *cleanliness* of the hotel are excellent. In the second case, only the *location* as-

Methods	Memory Usage (MB)	Training Time (minutes/epoch)	Appearance		Aroma		Palate	
			ValAcc	F1	ValAcc	F1	ValAcc	F1
multi-aspect collaborative training	24209	34.5	**89.2**	92.2	88.4	90.1	84.0	79.2
multi-task training	**19877**	**25.8**	**89.2**	**92.5**	**89.1**	**92.0**	**84.7**	**82.0**

Table 7: Ablation study on different training strategies.

pect appeared in the text. Correspondingly, MARE did not select any rationale other than the *location* aspect. This indicates that MARE benefits from multi-aspect collaborative encoding and makes decisions when there is clear evidence.

6.2 Ablation Studies

To verify the effectiveness of our model components, we have conducted several ablation studies on the BeerAdvocate dataset (McAuley et al., 2012).

6.2.1 multi-task training v.s. multi-aspect collaborative training

To explore the impact of multi-task training on the model as described in Section 4.3, this experiment verifies the effectiveness of multi-task training by comparing the performance, memory usage, and time cost of multi-task training and multi-aspect collaborative training.

The experimental result is shown in Table 7. The performance of multi-task training is slightly better than that of multi-aspect collaborative training. This is because, in the early stages of training, MARE cannot distinguish various aspects well, so multi-aspect collaborative training may lead to information leakage between different aspects, resulting in a performance drop. Meanwhile, multi-aspect collaborative training requires mask calculation for all aspects, resulting in high memory usage and long training time, reaching 24209MB and 34.5 minutes respectively. By contrast, multi-task training only requires encoding a single aspect at a time, so it costs much lower in both memory and training time. It saves 17.9% and 25.2% of memory usage and training time, respectively. This indicates that models trained using multi-task training can outperform those trained using multi-aspect collaborative training with fewer computational resources, demonstrating the effectiveness of multi-task training.

6.2.2 Hard Deletion v.s. Attention Mask Deletion

To demonstrate the effectiveness of hard deletion, this section contrastively employs AMD operations in the MAC. Specifically, we will replace the Equation (5)-(8) with Equation (16)-(19):

$$\mathbf{m}' = \sum_{i=0}^{k-1} \mathbf{m}[i] \in [0, k]^L \tag{16}$$

$$\tilde{m}[i] = \begin{cases} 0, \text{ If } m'[i] = 0 \\ 1, \text{ Otherwise} \end{cases} \tag{17}$$

$$\hat{\mathbf{m}} = \mathbf{m}' - \text{StopGrad}(\mathbf{m}') + \tilde{\mathbf{m}} \in \{0, 1\}^L \tag{18}$$

$$\tilde{\mathbf{A}}_i^h = \mathbf{A}_i^h \odot \hat{\mathbf{m}}, \text{ for } h \text{ in } 1, 2, ..., H \tag{19}$$

, where k means the number of aspects, and $\mathbf{m}'$ represents the mask vector with a span of closed interval [0, k], $\hat{\mathbf{m}}$ indicates the calculated mask vector to multiply with attention score matrix. Here, we also use the Straight Through technique to bypass the non-differentiable problem.

Experimental results are shown in Table 8. While using AMD, the rationalization and downstream performance are very poor. On the contrary, MARE-hard performs very well. In three aspects, the validation accuracy of MARE-hard was very close to BERT, and exceeded MARE-AMD by 3.5%, 4.8%, and 6.3%, respectively. Meanwhile, MARE-hard leads MARE-AMD by 23.1%, 23.4%, and 78.1% in rationalization performance, respectively. The reason is that AMD fails to effectively separate tokens corresponding to different aspects, leading to information leakage and hindering accurate rationale extraction. This indicates that AMD is not suitable for multi-aspect collaborative coding, and also proves the necessity and effectiveness of using hard deletion.

Methods	Appearance		Aroma		Palate	
	ValAcc	F1	ValAcc	F1	ValAcc	F1
BERT	**90.2**	-	**89.5**	-	**86.8**	-
MARE-AMD	85.7	69.4	84.3	68.6	78.4	3.9
MARE-hard	89.2	**92.5**	89.1	**92.0**	84.7	**82.0**

Table 8: Ablation study on different delete methods.

6.2.3 Special Token Initialization

To evaluate the impact of different initialization methods for special tokens on the model perfor-

mance, this section explores three distinct initialization approaches:

- random initialization: The first special token is initialized by [CLS], while all other special tokens are randomly initialized.
- CLS initialization: All the special tokens are initialized by [CLS].
- sharing initialization: All the special tokens are shared and initialized by [CLS].

The performance comparisons are shown in Table 9. MARE-CLS is slightly better than MARE-random and the MARE-share performs the worst. We found that MARE share cannot distinguish the differences in sparsity between different aspects. MARE-CLS achieves the best performance because the special token [CLS] is a highly informative embedding after pre-training. By default, MARE uses the CLS initialization.

Methods	Appearance		Aroma		Palate	
	ACC	F1	ACC	F1	ACC	F1
MARE-random	**85.7**	87.1	85.4	90.7	87.0	80.9
MARE-share	**85.7**	85.1	84.3	88.1	87.1	79.0
MARE-CLS	85.6	**92.5**	**86.0**	**92.0**	**88.0**	**82.0**

Table 9: Ablation study on different initialization strategies.

7 Conclusion

This paper proposed a Multi-Aspect Rationale Extractor to solve the limitations of traditional uni-aspect encoding models. MARE can collaboratively predict and interpret multiple aspects of text simultaneously. Additionally, MARE incorporated multi-task training, sequentially training on data from each aspect, thereby significantly reducing training costs. Extensive experimental results on two unsupervised rationale extraction datasets have shown that the rationalization performance of MARE is superior to all previous models. Ablation studies further demonstrated the effectiveness of our method.

Limitations

All of the above experiments have demonstrated the effectiveness of our method, but there are some limitations. MARE needs to prepend some special tokens in front of the input, which increases the computational overhead. Meanwhile, MARE can only adapted in encoder-based pre-trained language models. We are working hard to apply it to decoder-only models so that MARE can explain the predictions of LLMs. We will try to eliminate these limitations in our future work.

Acknowledgements

This work was supported in part by the National Key Research and Development Program of China (No.2021YFF1201200), the Science and Technology Major Project of Changsha (No.kh2402004). This work was carried out in part using computing resources at the High-Performance Computing Center of Central South University.

References

Diego Antognini, Claudiu Musat, and Boi Faltings. 2021. Multi-dimensional explanation of target variables from documents. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 12507–12515.

Yujia Bao, Shiyu Chang, Mo Yu, and Regina Barzilay. 2018. Deriving machine attention from human rationales. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1903–1913, Brussels, Belgium. Association for Computational Linguistics.

Jasmijn Bastings, Wilker Aziz, and Ivan Titov. 2019. Interpretable neural predictions with differentiable binary variables. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2963–2977, Florence, Italy. Association for Computational Linguistics.

Shiyu Chang, Yang Zhang, Mo Yu, and Tommi Jaakkola. 2019. A game theoretic approach to class-wise selective rationalization. *Advances in neural information processing systems*, 32.

Shiyu Chang, Yang Zhang, Mo Yu, and Tommi Jaakkola. 2020. Invariant rationalization. In *International Conference on Machine Learning*, pages 1448–1458. PMLR.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the NAACL 2019: Human Language Technologies, Volume 1 (Long and Short Papers)*, Minneapolis, Minnesota.

Yongfeng Huang, Yujun Chen, Yulun Du, and Zhilin Yang. 2021. Distribution matching for rationalization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 13090–13097.

Han Jiang, Junwen Duan, Zhe Qu, and Jianxin Wang. 2023. You only forward once: Prediction and rationalization in a single forward pass. *arXiv preprint arXiv:2311.02344*.

Yoon Kim. 2014. Convolutional neural networks for sentence classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Doha, Qatar.

Tao Lei, Regina Barzilay, and Tommi Jaakkola. 2016. Rationalizing neural predictions. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 107–117, Austin, Texas. Association for Computational Linguistics.

Wei Liu, Haozhao Wang, Jun Wang, Zhiying Deng, Yuankai Zhang, Cheng Wang, and Ruixuan Li. 2024a. Enhancing the rationale-input alignment for self-explaining rationalization. In *2024 IEEE 40th International Conference on Data Engineering (ICDE)*, pages 2218–2230. IEEE.

Wei Liu, Haozhao Wang, Jun Wang, Ruixuan Li, Xinyang Li, Yuankai Zhang, and Yang Qiu. 2023a. Mgr: Multi-generator based rationalization. *arXiv preprint arXiv:2305.04492*.

Wei Liu, Haozhao Wang, Jun Wang, Ruixuan Li, Chao Yue, and YuanKai Zhang. 2022. Fr: Folded rationalization with a unified encoder. In *Advances in Neural Information Processing Systems*, volume 35, pages 6954–6966. Curran Associates, Inc.

Wei Liu, Jun Wang, Haozhao Wang, Ruixuan Li, Zhiying Deng, Yuankai Zhang, and Yang Qiu. 2024b. D-separation for causal self-explanation. *Advances in Neural Information Processing Systems*, 36.

Wei Liu, Jun Wang, Haozhao Wang, Ruixuan Li, Yang Qiu, YuanKai Zhang, Jie Han, and Yixiong Zou. 2023b. Decoupled rationalization with asymmetric learning rates: A flexible lipschitz restraint. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, KDD '23, page 1535–1547, New York, NY, USA. Association for Computing Machinery.

Ilya Loshchilov and Frank Hutter. 2017. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*.

Julian McAuley, Jure Leskovec, and Dan Jurafsky. 2012. Learning attitudes and attributes from multi-aspect reviews. In *2012 IEEE 12th International Conference on Data Mining*, pages 1020–1025. IEEE.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. 2019. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32.

Hongning Wang, Yue Lu, and Chengxiang Zhai. 2010. Latent aspect rating analysis on review text data: a rating regression approach. In *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 783–792.

Ronald J Williams. 1992. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Reinforcement learning*, pages 5–32.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, et al. 2019. Huggingface's transformers: State-of-the-art natural language processing. *arXiv preprint arXiv:1910.03771*.

Mo Yu, Shiyu Chang, Yang Zhang, and Tommi Jaakkola. 2019. Rethinking cooperative rationalization: Introspective extraction and complement control. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 4094–4103, Hong Kong, China. Association for Computational Linguistics.

Mo Yu, Yang Zhang, Shiyu Chang, and Tommi Jaakkola. 2021. Understanding interlocking dynamics of cooperative rationalization. *Advances in Neural Information Processing Systems*, 34:12822–12835.

Linan Yue, Qi Liu, Li Wang, Yanqing An, Yichao Du, and Zhenya Huang. 2023. Interventional rationalization. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 11404–11418, Singapore. Association for Computational Linguistics.

A Implementation Details

A.1 Main Experiments

In the experiment, we utilize the Pytorch (Paszke et al., 2019) deep learning framework and the huggingface transformers library (Wolf et al., 2019) to implement MARE. BERT (Devlin et al., 2019) will be deployed as the backbone in MARE. MARE uses the AdamW optimizer (Loshchilov and Hutter, 2017) to optimize parameters, with a learning rate set to 3×10^{-5} and a weight decay set to 0.0. To control the sparsity and continuity of the generated rationales, this paper applies the "Cliff" deletion strategy, where the k is fixed at 9. In addition, we use grid search to select the most suitable hyperparameters β and γ from the candidate set $\{0.7, 1, 3, 5, 7\}$. We assume $\beta = \gamma$ in our experiments and select $\beta = \gamma = [0.7, 3, 3]$ for the BeerAdvocate dataset and Hotel Review dataset, respectively. During the training process, we adopt a balanced round-robin method to iteratively sample

data from all aspects. Set the batch size to 64 and limit the maximum sequence length to 256. For the BeerAdvocate dataset, MARE was trained for 15 epochs. However, considering the large scale of the Hotel Review dataset, the model only iteratively trained 5 epochs.

A.2 Implementation Details of Token Deletion

After obtaining the mask $\tilde{M}$ and its binarized counterpart $\tilde{\mathbf{M}}$ shown in Equation 5 and 6, we indeed multiply it with the attention score matrix to implement the token deletion. Specifically, the implementation in PyTorch is as follows:

```
def multi_head_attention(..., M, M_):
    ...

    att_score = Q @ K.transpose(-1, -2)
        / math.sqrt(d_head)

    # for token deletion
    M_grad = M + M_ - M_.detach()
    deleted_att_score = M_grad * torch.
        softmax(att_score, dim=-1)

    return deleted_att_score @ V
```

Listing 1: Token Deletion

This implementation achieves the following:

- The binary mask $\tilde{\mathbf{M}}$ determines which token pairs can interact (value 1) and which cannot (value 0).
- Multiplying $\mathbf{M\_grad}$ with the attention score matrix (att_score) effectively zeroes out attention scores between tokens of different aspects.
- The resulting attention scores are then used to compute the weighted sum of value vectors ($\mathbf{V}$).

This approach ensures that tokens within the same aspect can interact through the attention mechanism, while interactions between tokens of different aspects are prevented. This aligns with our goal of allowing aspect-specific information to be aggregated separately.

B Cliff Decay

The Cliff decay strategy is defined as follows:

- For layers $i < x$: All tokens are retained.
- For layers $i \geq x$: A proportion p of tokens are deleted.

Here, x is the layer at which deletion begins, and p is the deletion proportion. In our experiments, we set $x = 9$, with p varying by aspect.

C Training Stability of MARE

We have conducted additional experiments with different seeds to validate our training stability. Table 10 shows the standard deviations of F1 scores across 3 different seeds. As shown in the table, MARE demonstrates good stability, particularly in the Appearance and Aroma aspects. We believe this stability is partly due to the multi-aspect nature of our model, which allows it to leverage internal correlations between different aspects.

Method	Appearance	Aroma	Palate
MARE	0.4	0.3	1.3

Table 10: The F1 standard deviations of MARE across 3 different seeds.

D Experiments on the Correlated BeerAdvocate Dataset

To evaluate the impact of spurious correlation on model performance, we also conduct experiments on the correlated BeerAdvocate dataset (McAuley et al., 2012). The overall performance is shown in Table 11. As we can see, MARE achieves state-of-the-art performance and is better than existing methods for a large margin. We attribute this to the effectiveness of collaborative coding, demonstrating that internal correlations can suppress spurious correlations.

Methods	S	Appearance				Aroma				Palate			
		ACC	P	R	F1	ACC	P	R	F1	ACC	P	R	F1
RNP(Lei et al., 2016)	10	-	32.4	18.6	23.6	-	44.8	32.4	37.6	-	24.6	23.5	24.0
HardKuma(Bastings et al., 2019)		-	53.6	28.7	37.4	-	29.3	25.9	27.3	-	7.7	6.0	6.8
INVRAT(Chang et al., 2020)		-	42.6	31.5	36.2	-	41.2	39.1	40.1	-	34.9	45.6	39.5
Inter-RAT(Yue et al., 2023)		-	66.0	46.5	54.6	-	55.4	47.5	51.1	-	34.6	48.2	40.2
MGR(Liu et al., 2023a)		80.5	87.5	51.7	65.0	89.7	78.7	52.2	62.8	86.0	65.6	57.1	61.1
YOFO (Jiang et al., 2023)		87.7	96.4	61.9	75.4	**92.7**	95.4	65.2	77.5	**91.9**	67.4	67.4	67.4
MARE(*ours*)		**88.9**	**99.0**	**62.2**	**76.4**	92.0	**97.5**	**66.2**	**78.9**	90.8	**81.6**	**72.8**	**77.0**
RNP(Lei et al., 2016)	20	-	39.4	44.9	42.0	-	37.5	51.9	43.5	-	21.6	38.9	27.8
HardKuma(Bastings et al., 2019)		-	64.9	69.2	67.0	-	37.0	55.8	44.5	-	14.6	22.3	17.7
INVRAT(Chang et al., 2020)		-	58.9	67.2	62.8	-	29.3	52.1	37.5	-	24.0	55.2	33.5
Inter-RAT(Yue et al., 2023)		-	62.0	76.7	68.6	-	44.2	65.4	52.8	-	26.3	59.1	36.4
MGR(Liu et al., 2023a)		85.6	76.3	83.6	79.8	89.6	64.4	81.3	71.9	89.3	47.1	73.1	57.3
YOFO (Jiang et al., 2023)		88.4	77.5	87.6	82.2	91.9	**78.7**	92.8	**85.2**	91.3	44.6	75.4	56.0
MARE(*ours*)		**90.6**	**81.4**	**92.4**	**86.6**	**92.1**	74.0	**95.0**	83.2	**91.9**	**47.3**	**88.0**	**61.5**
RNP(Lei et al., 2016)	30	-	24.2	41.2	30.5	-	27.1	55.7	36.4	-	15.4	42.2	22.6
HardKuma(Bastings et al., 2019)		-	42.1	82.4	55.7	-	24.6	57.7	34.5	-	21.7	49.7	30.2
INVRAT(Chang et al., 2020)		-	41.5	74.8	53.4	-	22.8	65.1	33.8	-	20.9	71.6	32.3
Inter-RAT(Yue et al., 2023)		-	48.1	82.7	60.8	-	37.9	72.0	49.6	-	21.8	66.1	32.8
MGR(Liu et al., 2023a)		88.5	57.2	93.9	71.1	91.6	45.8	87.4	60.1	89.3	27.3	66.5	38.7
YOFO (Jiang et al., 2023)		**88.9**	63.5	94.3	75.9	92.4	53.6	88.7	66.8	91.6	34.0	75.7	46.9
MARE(*ours*)		88.7	**65.9**	**96.8**	**78.4**	**92.9**	**55.2**	**91.9**	**69.0**	**92.7**	**35.7**	**79.0**	**49.2**

Table 11: The results of different methods on correlated BeerAdvocate Dataset (McAuley et al., 2012).