E-VarM: Enhanced Variational Word Masks to Improve the Interpretability of Text Classification Models

Ling Ge[1], Chunming Hu[1,2,*], Guanghui Ma[1], Junshuang Wu[4], Junfan Chen[1], Jihong Liu[5], Hong Zhang[3], Wenyi Qin[1], Richong Zhang[1]
[1] School of Computer Science and Engineering, Beihang University, Beijing, China
[2] College of Software, , Beihang University , Beijing, China
[3] National Computer Network Emergency Response Technical Team, Beijing, China
[4] Beijing Jinghang Research Institute of Computing and Communication, Beijing, China
[5] School of Mechanical Engineering and Automation, Beihang University, Beijing, China
{geling, hucm}@buaa.edu.cn, {magh, chenjf, zhangrc}@act.buaa.edu.cn,
zhangh@isc.org.cn, ryukeiko@buaa.edu.cn,wenyi.qin@hotmail.com

Abstract

Enhancing the interpretability of text classification models can help increase the reliability of these models in real-world applications. Currently, most researchers focus on extracting task-specific words from inputs to improve the interpretability of the model. The competitive approaches exploit the Variational Information Bottleneck (VIB) to improve the performance of word masking at the word embedding layer to obtain task-specific words. However, these approaches ignore the multi-level semantics of the text, which can impair the interpretability of the model, and do not consider the risk of representation overlap caused by the VIB, which can impair the classification performance. In this paper, we propose an enhanced variational word masks approach, named E-VarM, to solve these two issues effectively. The E-VarM combines multi-level semantics from all hidden layers of the model to mask out task-irrelevant words and uses contrastive learning to readjust the distances between representations. Empirical studies on ten benchmark text classification datasets demonstrate that our approach outperforms the SOTA methods in simultaneously improving the interpretability and accuracy of the model.

1 Introduction

With the widespread adoption of neural networks in text classification tasks (Bastings and Filippova, 2020; Halder et al., 2020; Schick et al., 2020; Lv et al., 2021), the classification models are expected to provide not only highly accurate classification results but also reasonable prediction rationales (Peake and Wang, 2018; Sun et al., 2021). These prediction rationales, which are short yet informative parts of the input for classification predictions (Bastings et al., 2019), manifest the interpretability of the model. The better the interpretability of the model, the more reasonable rationales the model provides (Lin et al., 2021). Therefore, improving the interpretability of the model helps to boost the reliability of these classifiers in real-world applications (Jacovi and Goldberg, 2020).

To improve the interpretability of the classifiers, many methods have been proposed (Chrysostomou and Aletras, 2021; Bastings et al., 2019). Some studies rely on additional inputs at the training time, such as pre-defined information and human annotations (Erion et al., 2019; Plumb et al., 2020), all of which involve high human costs. Other works resort to assigning a binary Bernoulli variable to each input word with promising results, among which the competitive approaches are the Sparse-VIB (Paranjape et al., 2020) and Vmask (Chen and Ji, 2020) models . They all employ the Variational Information Bottleneck (VIB) (Alemi et al., 2017) to train stochastic masks to automatically learn task-specific words, namely prediction rationales, for prediction and interpretability simultaneously.

However, these methods only utilize the information from the word embedding layer , which holds little task-specific information (Van Aken et al., 2019), ignoring the multi-level information of the text, such as syntactic and semantic information. That is, this initial layer results in poor interpretability performance and reaches low accuracy on related semantic tasks . Additionally, these approaches do not consider the risk of representation overlap arising from VIB (Alemi et al., 2017), resulting in the model failing to learn discriminative class representations, thus further impairing

*Corresponding Author

Proceedings of the 29th International Conference on Computational Linguistics, pages 1036–1050
October 12–17, 2022.

the classification performance.

To address the above issues, we propose an **E**nhanced **Var**iational Word **M**asks approach, (named E-VarM), which combines multi-level information to learn task-specific words in an unsupervised manner and uses contrastive learning to alleviate the representation overlap problem caused by VIB. As a result, both the interpretability and accuracy of the model are improved.

Specifically, as the data flow from the input to the output layer, each hidden layer of the text encoder can capture different information (Geirhos et al., 2019). For example, the BERT (Devlin et al., 2018) can encode the captured information to a rich hierarchy of linguistic information (Hewitt and Manning, 2019; Li et al., 2022) with surface features at the bottom, syntactic features in the middle, and semantic features at the top (Jawahar et al., 2019; van Aken et al., 2019; Kim et al., 2020; Gupta et al., 2022). Inspired by the feature extraction mechanism of the text encoder, we fuse multilevel information from all hidden layers of the model to generate a task-specific binary Bernoulli distribution to extract reasonable interpretations for various text classification tasks, such as sentiment analysis, syntactic judgments, and semantic inference.

Furthermore, the VIB-based methods rely on the Lagrangian factor to perform the compression-prediction trade-off (Tishby et al., 2000; Mahabadi et al., 2021). A larger Lagrangian factor means masking out more task-irrelevant information (Pan et al., 2021; Kolchinsky et al., 2019; Gálvez et al., 2020) for a special class and obtaining better interpretability. However, along with this larger value is the potential risk of representation overlap or even collapse (Alemi et al., 2017; Goldfeld and Polyanskiy, 2020; Wu et al., 2020) since it may hinder the model from learning the discriminative representation and lead to loss of class information. Therefore, we leverage supervised contrastive learning (Khosla et al., 2020) to adjust the classification representation by pulling in samples from the same class and pushing away samples from different classes, thus mitigating the representations overlap problem. Additionally, we resort to task-specific words to construct diverse positive samples to enhance the efficiency of contrastive learning.

In a nutshell, we make the following major contributions: (1) We introduce an enhanced variational word masks method to improve classification performance and interpretability simultaneously. (2) Our method produces token-level interpretations that consider the multi-level information from all model layers and are adaptable to various classification tasks. (3) To the best of our knowledge, our approach is the first attempt to use contrastive learning to alleviate the representation overlap caused by VIB. (4) The experimental results on ten benchmark datasets validate the effectiveness of our method.

2 Related Works

2.1 Model Interpretability

Various approaches have been proposed to improve the interpretability of neural networks, such as exploiting attention distribution on tokens (Sun et al., 2020), extracting subsets of input text (Swanson et al., 2020), or relying on language models to generate explanations from scratch (Rajani et al., 2019). The underlying techniques in this paper are closely related to the extracted interpretation methods. Some of these works leverage pre-collected annotations (Erion et al., 2019; Plumb et al., 2020), which can be labor-intensive. Other works use random masks on the input to automatically learn interpretable models, for example, Bastings et al. (2019) and Cao et al. (2020) adopt $L0$ constraints to make the masks sparse.

The $L0$ regularisation overemphasizes the sparsity, which can damage the accuracy of model. Therefore, some researchers introduce VIB to control the level of mask sparsity through a tunable sparse prior (Paranjape et al., 2020). Among these, the competitive methods are the SparseVIB (Paranjape et al., 2020), VIBI (Bang et al., 2021), and Vmask (Chen and Ji, 2020). However, these methods only consider the information of word embeddings to extract task-specific tokens and ignore the risk of representation overlap, which affects the models' interpretability and classification performance. This paper aims to address the above issues.

2.2 Supervised Contrastive Learning

Supervised contrastive learning performs representation learning by expanding the embedding differences of instances from different classes in the hidden space (Khosla et al., 2020), which is widely used in sentiment recognition (Liang et al., 2021) , semantic inference (Zhang et al., 2021) and text classification (Gunel et al., 2021) with good results. Previous works (Wu et al., 2021; Robinson et al., 2021) indicate that constructing diverse posi-

tive and negative samples facilitates discriminative feature learning. The reason is that high-quality contrastive samples encourage the model to mine both intra-class and inter-class features, forcing the model to perform fine-grained representation learning (Khosla et al., 2020). Therefore, we use contrastive learning to adjust representations to mitigate the risk of overlap and create high-quality positive samples based on task-specific words to enhance the model's classification performance.

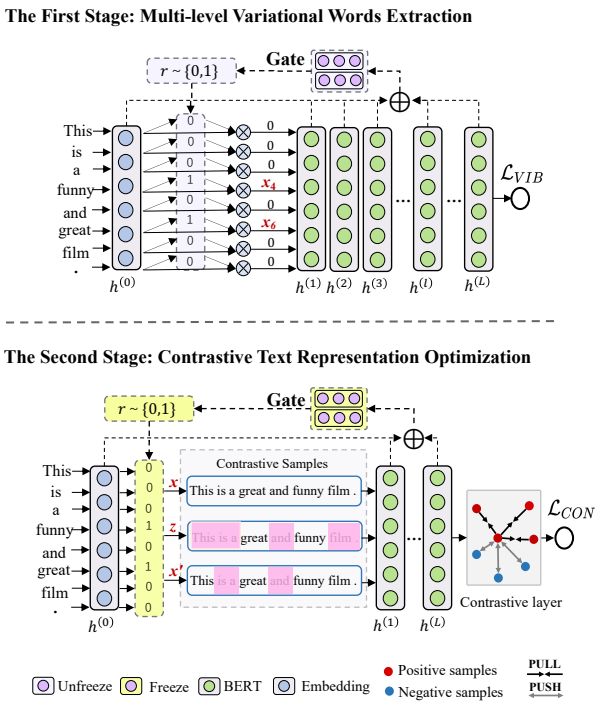


Figure 1: An illustration of our proposed method. The x, z, and x' are contrastive samples in the form of word embeddings. We use tokens to replace embeddings to better indicate the diversity among samples. In this case, the input sentence is from the sentiment analysis dataset and labeled as positive.

3 Methodology

3.1 Overview

This paper aims to simultaneously improve the interpretability and accuracy of the text classifier. Our model consists of two stages shown in Fig 1. The first one is 1) the multi-layer variational words extraction, which automatically extracts task-specific words as rationales. Another is 2) the contrastive text representation optimization, which mitigates the feature overlap of different classes and learns discriminative representations.

To optimize these two stages, we adopt an iterative training mechanism at the batch level for the whole model. With this mechanism, the model can be corrected in time if the representation overlap occurs in the extraction stage.

3.2 Multi-level Variational Words Extraction

The core idea of this stage is that if a subset of an input text can be removed without affecting the prediction, this subset text is considered task-irrelevant, while the remaining subset text is task-specific. Generally, this stage aims to learn a sparse random gate g_ϕ(a.k.a., masks) for a neural text classifier $f_\theta(\cdot)$ to obtain task-specific words. In this paper, we use BERT as the text classifier.

Specifically, given an input $x = \{x_1, x_2, ..., x_n\}$, where $x_t \in \mathbb{R}^d$ indicates the word embedding, we feed it into the encoder to obtain the hidden states $\{h^{(0)}, ..., h^{(L)}\}$ of different layers of BERT. In this hierarchy of linguistic information structure encoded by BERT, surface features are at the bottom, syntactic features are in the middle, and semantic features are at the top. Herein, we stack the hidden states up to $h(L)$ as input to the gate network , a two-hidden-layer MLP, to predict the binary output r,

$$r = g(h^{(0)} \oplus h^{(1)} ... \oplus h^{(L)}) \tag{1}$$

where $\oplus$ denotes concatenate operation. We rely on $r \in \mathbb{R}^{2n}$ to perform word masks, so that the model has multi-level information and is adaptable to various tasks. The r_i associated with x_i follows the Bernoulli distribution,

$$r_i \sim Bernoulli(\alpha_i) \tag{2}$$

where α_i is the probability of the word embedding x_i being selected as the task-specific word. Precisely, the binary r produced by the gate can be expressed as the corresponding two-element one-hot vector $\boldsymbol{r_i} = [r_{i,j}]_{j=0,1}$, where $r_{i,j} = 1$ means $r_i = j$. Similarly, $\alpha_{i,j}$ is the probability that $r_i = j$ (Xue et al., 2020),

$$\begin{aligned} \boldsymbol{r_i} &= \text{one\_hot}(\arg\max_{\mathrm{j}} \alpha_{\mathrm{i,j}}, \mathrm{j} = 0, 1) \\ \alpha_{i,0} &= 1 - \alpha_i, \alpha_{i,1} = \alpha_i \end{aligned} \tag{3}$$

However, sampling from the Bernoulli distribution (Equation 2) causes the non-differentiability problem (Xue et al., 2020). We adopt the Gumbel-Softmax distribution (Xue et al., 2020; Jang et al., 2017) to approximate r_i differentiably,

$$\hat{\boldsymbol{r}}_{\boldsymbol{i}} = [\hat{\alpha}_{i,j}]_{j=0,1}$$
$$\hat{\alpha}_{i,j} = \frac{exp((\log(\alpha_{i,j}) + \epsilon_j)/\tau)}{\sum_{k=0}^{1} exp((log(\alpha_{i,k}) + \epsilon_k)/\tau)} \quad (4)$$

where ϵ_j is sampled randomly from Gumbel(0, 1), and the temperature τ controls how closely the Gumbel-Softmax distribution approximates the one-hot. By replacing Equation 2 with $r_i = \hat{r}_{i,0}$, we can train the model in an end-to-end manner.

Finally, we obtain the task-specific word embeddings z as the final input for prediction, i.e.,

$$z = r \odot x \quad (5)$$

where $\odot$ is an element-wise multiplication and z is a subset of x.

Since there is no direct supervision signal to estimate the gate parameters ϕ, we follow the standard practice in the Information Bottleneck(IB) (Tishby et al., 2000) theory to optimize the parameters. For the input x and its label y, the IB principle aims to learn the minimal sufficient representation z that preserves enough information about the output y (prediction) while containing the least redundant information from input x (compression) (Alemi et al., 2017; Mahabadi et al., 2021),

$$\mathcal{L}_{IB} = \ -I(z, y) + \beta \cdot I(x, z) \quad (6)$$

where β is the Lagrangian factor to balance the compression and prediction, and $I(.,.)$ is the mutual information.

To compute the two mutual information items above, the Deep VIB(Alemi et al., 2017) perform a variational approximation for the IB objective via a neural network. Thus, we obtain $\mathcal{L}_{VIB}$, which is the approximation of $\mathcal{L}_{IB}$,

$$\mathcal{L}_{VIB} = -\sum_{i=0}^{n} p(z|x^{(i)}) \log q(y^{(i)}|\boldsymbol{z}) + \beta \cdot \boldsymbol{KL}[p(z|x^{(i)})||m(z))] \quad (7)$$

where $q(y^{(i)}|z)$ is a parametric approximation of $p(y^{(i)}|z)$, $m(z)$ is a variational estimate of the prior probability $p(z)$ of z, and $p(z|x^{(i)})$ is an estimate of the posterior probability of z. The $\boldsymbol{KL}[\cdot||\cdot]$ denotes Kullback-Leibler divergence, and the β is inherited from the IB theory.

Since the compressed features z is determined by the random variable r that follows the Bernoulli distribution (Equation 5), we can rewrite Equation 7 with r. Motivated by VMask (Chen and Ji, 2020) and mean-field approximation (Tanaka, 1998), we obtain $m(r) = \Pi_{i=1}^{n} m(r_i)$. As $m(r_i) = Bernoulli(0.5)$ means that each word has an equal probability to be masked or selected (Chen and Ji, 2020), we get the uniform distribution $m(r)$ and Equation 7 can be further simplified as,

$$\mathcal{L}_{VIB} = -\sum_{i=0}^{n} p(r|x^{(i)}) \log q(y^{(i)}|r, x^{(i)}) + \beta \cdot \sum_{i=0}^{n} p(r|x^{(i)}) \log p(r|x^{(i)}) \quad (8)$$

The first term in $\mathcal{L}_{VIB}$ is a cross-entropy aiming to make sure the information in $p(r|x)$ for predicting is sufficient. The second term in $\mathcal{L}_{VIB}$ is to regularize $p(r|x)$ to make masks sparse, enabling the r vector to contain more zeros. To compute the compressed posterior $p_{\theta,\phi}(r|x^{(i)})$, we first feed input embedding x to BERT model $f_\theta(\cdot)$ and then resort to the amortization network (Rezende and Mohamed, 2015; Chen and Ji, 2020) , which is our gate network g_ϕ, to output binary value r. Then the BERT takes r along with $x^{(i)}$ as input and produces a probability of output $y^{(i)}$, $q_\theta(y^{(i)}|r, x^{(i)})$.

In actual training, we use the $\mathcal{L}_{VIB}$ to simultaneously optimize the gate parameter ϕ and the classifier parameters θ.

3.3 Contrastive Text Representation Optimisation

In this stage, we leverage supervised contrastive learning to mitigate the risk of representation overlap.

Specifically, for a given input x, we obtain a compressed sample z that contains only the task-specific embeddings through the first stage. To perform supervised contrastive learning, we treat the samples belonging to the same class in a batch as positive samples and the rest within the batch as negative samples. Additionally, to increase the diversity of the contrastive samples, we introduce additional positive samples x',

$$x' = x - R(x - z) \quad (9)$$

where $R(x - z)$ denotes a random mask operation on task-irrelevant embeddings, and the number of masks ranges from 1 to m, with m being the number of class-irrelevant words of input x. Since we

Table 1: Statistics of the datasets. "Class" is the number of labels. "Ave.Len" represents the average length of a sentence. "Train", "Dev" and "Test" denote the size of the training set, validation set and test set, respectively.

Num	Dataset	Class	Ave.Len	Train	Dev	Test
1	QNLI	2	40	104K	5463	-
2	QQP	2	23	363K	40k	-
3	COLA	2	8	8551	1043	-
4	IMDB	2	268	25K	-	25K
5	AGNews	4	32	120K	-	7.6K
6	Yelp	2	138	560K	-	38K
7	Subj	2	23	10K	-	-
8	RT	2	23	10K	-	-
9	SST-1	5	18	8540	1101	2208
10	SST-2	2	19	6920	872	1821

deliberately avoid a mask on task-specific features when perturbing the input x, the label of x' is not changed. These diverse contrastive samples, x, x' , and z, can force the model to focus on task-specific words and increase contrastive learning efficiency.

Then, we freeze the gate network and use the above samples to optimize text representations. Given a sample representation h_i , its positive representation h_j and negative representation h_k , the supervised contrastive learning loss is presented as,

$$\mathcal{L}_{CON} = \sum_{i=1}^{3N} \frac{-1}{3N_{\tilde{y}_i} - 1} \sum_{j=1}^{3N} \mathbb{I}_{i \neq j} \cdot \mathbb{I}_{\tilde{y}_i = \tilde{y}_j} \cdot log \frac{exp(h_i \cdot h_j / \tau)}{\sum_{k=1}^{3N} \mathbb{I}_{i \neq k} \cdot exp((h_i \cdot h_k / \tau))} \tag{10}$$

where $i, j, k \in \{1, 2, \cdots, 3N\}$ and these 3N samples consist of N x samples, N x' samples, and N z samples. The τ is a temperature controlling the concentration level of the distribution (Hinton et al., 2015).

Finally, we obtain the contrastive loss $\mathcal{L}_{CON}$ for the second stage to fine-tune the BERT encoder.

4 Experimental Settings

4.1 Dataset

We adopt ten benchmark datasets to evaluate our model, ranging from sentiment classification, and syntactic judgment to semantic inference, etc., which includes five sentiment analysis datasets: **RT** (Pang and Lee, 2005a), **IMDB** (Maas et al., 2011), **SST-2** (Socher et al., 2013), **Yelp** (Zhang et al., 2015), **SST-1** (Socher et al., 2013); two semantic inference datasets: **QQP** (Wang et al., 2018), **QNLI** (Wang et al., 2018); one topic categorization dataset:**AG's News** (Zhang et al., 2015); one grammatical judgment dataset: **COLA** (Wang et al., 2018); and one subjective / objective classification dataset: **Subj** (Pang and Lee, 2005b). The statistics of the datasets are displayed in Table 1.

Since QNLI, QQP, and COLA datasets have no test set, we randomly select 20% of the training set from each of them as their respective test set. Similarly, for IMDB, AGNews, and Yelp datasets, we randomly selecte 20% of the training data as their respective validation set. For Subj and RT datasets, which have no both validation and test set, we randomly select 10% of their respective training data as their validation and test set, respectively. More details about the datasets can be found in the **Appendix A**.

4.2 Baseline and SOTAs

We conduct experiments with seven competitive models, one baseline model: **BERT**$_{Base}$ (Devlin et al., 2018), three SOTA models, and three variants of our model.

AGN (Li et al., 2021): An adaptive gate-based SOTA model that improves performance by computing corpus-specific features such as word frequency and label distribution.

SCL (Gunel et al., 2021): A contrastive learning-based SOTA model that utilizes supervised contrastive learning to obtain promising features and enhance performance.

Vmask (Chen and Ji, 2020): An interpretation-based SOTA model that improves accuracy and interpretation by masking task-irrelevant words at the word embedding layer with the VIB.

E-VarM$_{MASK}$: A variant of E-VarM that removes the representation optimization stage and contains only the variational word extraction stage.

E-VarM$_{\text{PPL}}$: A variant of E-VarM that uses a two-stage pipeline training method instead of the iterative one. We first train the gate and the Bert jointly for 50 epochs to extract class-specific words. Then we fix the gate parameters and tune the Bert for 50 epochs via contrastive learning.

E-VarM$_{\text{ETE}}$: The third variant of E-VarM that uses an end-to-end training method to update both the gate and the BERT model. The corresponding loss function $\mathcal{L}_{ETE}$ can be expressed formally as,

$$\mathcal{L}_{ETE} = \mathcal{L}_{VIB} + \mathcal{L}_{CON} \tag{11}$$

where $\mathcal{L}_{VIB}$ is the VIB loss shown in Equation

Table 2: The prediction accuracy (%) comparison of different methods. The best results are marked in bold.

Methods	QNLI	QQP	COLA	IMDB	AGNews	Yelp	Subj	RT	SST-1	SST-2
BERT_{Base}	87.07	90.17	82.45	91.75	93.09	96.30	96.40	86.59	50.81	90.44
AGN	86.95	90.23	81.20	92.66	93.59	97.00	95.61	85.56	50.70	90.57
SCL	86.57	89.26	82.46	91.78	93.61	96.40	95.33	86.65	50.59	90.72
Vmask	85.03	86.04	79.41	92.06	93.52	96.30	96.68	87.82	51.99	91.21
E-VarM	**87.94**	**91.08**	**83.99**	**92.73**	**94.00**	**97.28**	**97.30**	**88.67**	**53.44**	**92.37**
Variants										
$\text{E-VarM}_{\text{MASK}}$	86.00	87.21	80.19	92.67	93.58	96.32	96.90	88.00	51.96	92.04
$\text{E-VarM}_{\text{ETE}}$	71.44	79.54	69.70	85.72	90.78	93.92	93.80	82.94	46.87	87.04
$\text{E-VarM}_{\text{PPL}}$	87.59	89.89	82.64	92.76	94.00	96.64	96.80	88.19	52.67	91.76

8 and $\mathcal{L}_{CON}$ is the contrastive loss shown in Equation 10.

4.3 Evaluation Metrics

In our experiments, we chose accuracy as evaluation metric for the classification performance. To assess the interpretability of different models, we follow the previous work (Chen and Ji, 2020) and choose Area of Perturbation Curve (AOPC) (Nguyen, 2018) and post-hoc accuracy (Chen et al., 2018) as the local interpretability and the global interpretability metrics, respectively.

The Local Interpretability (AOPC) : The AOPC metric calculates the average change in prediction probability over all classes by removing the top K most important words from the input. To provide a fair assessment for all compared methods, we utilize LIME (Ribeiro et al., 2016) to extract the nine most important words. Specifically, the LIME is a local interpretable algorithm that can extract local explanations for a classifier by fitting the local decision boundary of an instance under test (Ribeiro et al., 2016).

The AOPE metric is then calculated as follows,

$$AOPC = \frac{1}{K+1}\langle \sum_{k=1}^{K} (p(\hat{y}|x) - p(\hat{y}|x_{\setminus 1\cdots k})) \rangle_{p(x)} \tag{12}$$

where $p(\hat{y}|x_{\setminus 1\cdots k})$ is the probability for the predicted class with words 1..K removed and $\langle\cdot\rangle_{p(x)}$ denotes the average over all examples.

The Global Interpretability (the post-hoc Accuracy) : The post-hoc accuracy assesses the sufficiency of important words selected from the input for model prediction and is calculated as follows,

$$ACC_{post-hoc}(k) = \frac{1}{N}\sum_{i=1}(\mathbb{I}[\hat{y}(x_i^{(k)}) = \hat{y}(x_i)]) \tag{13}$$

where $\mathbb{I}[\cdot]$ is an indicator function. $\hat{y}(x_i)$ represents the prediction label of sample x_i and $\hat{y}(x_i^{(k)})$ means the prediction label obtained using the most important k words from x_i.

4.4 Implementation Details

We use the pre-trained BERT (Wolf et al., 2019) as the classifier and adopt a batch-level iterative mechanism to train the model, with each iteration consisting of two stages. In the first stage, we train the gate and BERT model jointly, and for the second stage, we freeze the gate and train only the BERT model. Both stages are performed using the Adam optimizer(Kingma and Ba, 2015) with learning rate = 1e-5, batch size = 64, and the dropout=0.2 empirically.

We utilize the grid search technology to obtain the optimal super-parameters, including the Lagrangian multiplier β and the contrastive learning temperature τ. The β is selected from {0.1,0.5, 1, 10, 50},while the τ is chosen from 0.1 to 0.9.

In order to compare the performance of the different models, We evaluate the AGN and Vmask using the open-source code [1] and [2] respectively. Since the source code for SCL is not provided, we implement and evaluate this method as described in the original paper. Additionally, our approach is implemented using PyTorch, and all calculations are done on NVIDIA Tesla V100 GPU, with per experiment taking approximately 1∼3 hours.

[1] https://github.com/4AI/AGN
[2] https://github.com/UVa-NLP/VMASK

Table 3: The AOPC accuracy(%) comparison of different methods. The best results are marked in bold.

k	Method	QNLI	QQP	RT	COLA	IMDB	SST1	SST2	AGNews	Subj	Yelp
1	Vmask	3.44	4.61	7.19	5.61	3.38	7.80	10.52	2.34	**2.31**	1.15
	E-VarM	**6.27**	**6.37**	**8.18**	**8.57**	**4.25**	**8.86**	**11.16**	**3.08**	1.94	**1.61**
5	Vmask	14.09	13.22	25.16	13.02	13.29	17.73	33.53	9.76	**13.75**	6.17
	E-VarM	**21.04**	**16.23**	**28.58**	**17.83**	**13.52**	**19.67**	**35.43**	**12.69**	13.14	**8.29**
9	Vmask	18.65	15.04	31.84	15.05	19.89	19.20	37.78	15.60	22.97	9.55
	E-VarM	**26.31**	**17.54**	**36.70**	**22.50**	**20.47**	**21.34**	**39.49**	**19.63**	**23.32**	**12.61**

5 Experimental Results

5.1 The Classification Performance

As shown in Table 2, our approach achieves the best results in most datasets, proving its strength in various text classification tasks. More specifically, we can draw the following conclusions.

Compared with the gate-based model. AGN exploits an adaptive gate to obtain data-specific prior distributions to boost its accuracy. However, the prior distribution of the data does not always contribute to the model's performance, leading to inferior results. In contrast, our method always concentrates on task-specific words and is not restricted by the prior distribution of the data.

Compared with the contrastive learning-based model. Our method has enhancements over SCL on all datasets due to the high-quality positive samples. These samples, created with the help of task-specific words, will increase the challenge of contrastive learning and facilitate the model to learn better class discriminative representations.

Compared with the interpretable-based model. Our approach outperforms Vmask on most datasets, in particular, with obvious improvements on the QNLI, QQP, and COLA datasets. Since Vmask only uses word embedding information to select important words, it has poor selection decisions on tasks that rely on mid-level or high-level information, such as semantic reasoning (QNLI, QQP) and syntactic judgment (COLA), decreasing the classification performance.

Compared with the variants. When removing the representation optimization stage, E-VarM$_{\mathrm{MASK}}$ shows a performance degradation on all datasets compared to E-VarM, illustrating that contrastive learning can improve prediction performance. Since the gate network of E-VarM$_{\mathrm{ETE}}$ is adversely affected by contrastive loss, the performance of E-VarM$_{\mathrm{ETE}}$ decreases substantially compared to E-VarM. As the encoder of E-VarM$_{\mathrm{PPL}}$ cannot be corrected in time if there is a representation overlap in the extraction stage, so that the performance of E-VarM$_{\mathrm{PPL}}$ slightly drops compared to E-VarM. In contrast, E-VarM uses a two-task iterative way at the batch level, where words extraction is performed and immediately followed by the representation adjustment within a batch.

5.2 The Interpretability

Because of the computational cost, we select 500 samples randomly from the test set to evaluate the interpretability of the models.

The local interpretability (AOPC). We remove k task-specific words from the input for the AOPC experiment (Equation 12) with k selected from $\{1, 5, 9\}$. As shown in Table 3 , the AOPC of E-VarM outperforms Vmask on almost all datasets. On the four datasets: AG's News, COLA, QQP, and QNLI, E-VarM shows a noticeable improvement over Vmask. For example, it outperforms Vmask by 7% at K=5 on QNLI. These four datasets involve complex tasks, such as topic classification, semantic inference, and grammar judgment, and rely on medium or high-level information to select task-specific words as explanations. E-VarM integrates multi-level information for words selection and has good interpretability for such complex tasks. Even for datasets that rely on shallow features to mask, such as SST2, and IMDB datasets, E-VarM can slightly outperform Vmask by about 2% on average.

The global interpretability (Post-hoc accuracy). To compute the post-hoc accuracy of models, we use the top k task-specific words from the input for prediction and compare it with the result from the whole input (Equation 13), with k ranging from 1 to 9. As shown in Fig 2, the global interpretability of E-VarM outperforms Vmask on almost all datasets, with the best performance on the COLA dataset. On datasets that require multi-level information for masking, such as QNLI, QQP,

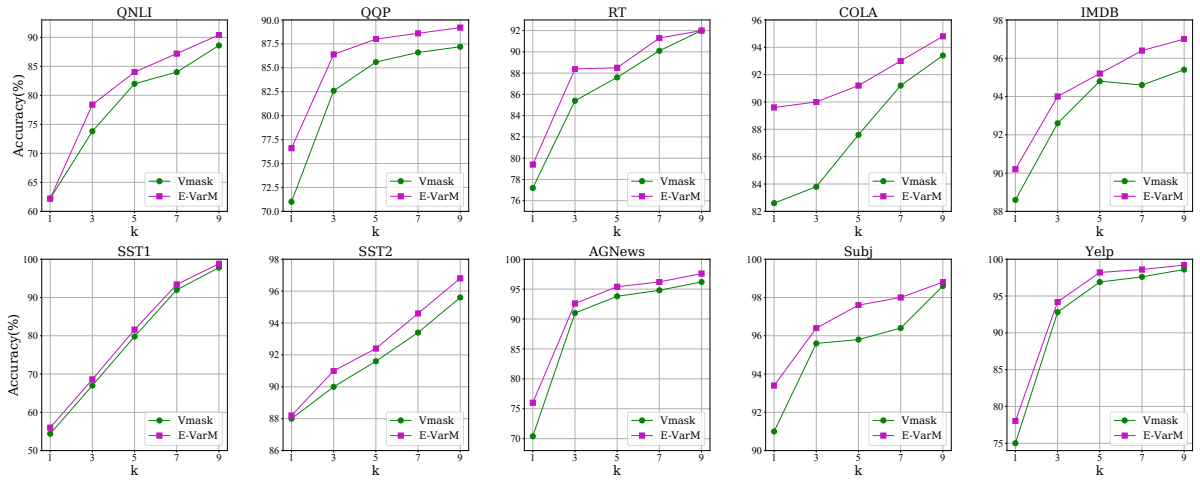
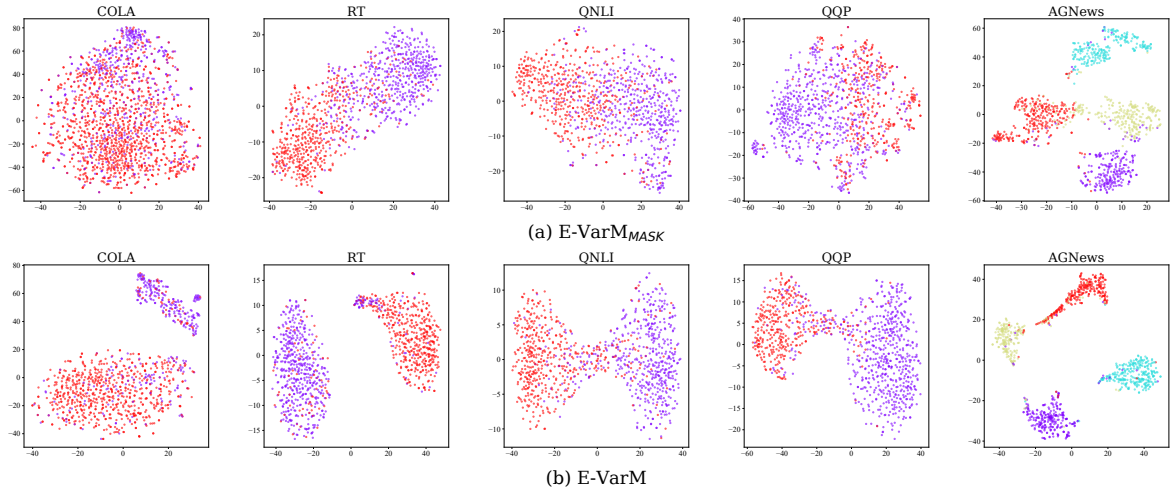


Figure 2: The post-hoc accuracy(%) comparison of different methods on ten datasets.



(a) E-VarM$_{MASK}$

(b) E-VarM

Figure 3: The TSNE visualization comparison of text representations of different models.

and RT, there is a gap between Vmask and E-VarM, which further illustrates the superiority of our model in handling complex tasks. Also, E-VarM achieves relatively good results on IMDB, a dataset with an average of 268 tokens per sentence, demonstrating the outstanding performance of our model in extracting task-specific words.

5.3 The Effect of Different Levels of Information.

We adopt the bottom (1∼4 layer), middle (5∼8 layer), and top (9∼12 layer) layers of information for model performance evaluation to demonstrate the importance of using multi-layer information when selecting task-specific words. As shown in Table 5, the model , relying only on low-layer information (low-layer model), achieves good performance when performing shallow-level tasks such as sentiment analysis. In contrast, when the model uses only middle or high-layer information (middle-layer or high-layer model), its performance decreases slightly compared to the low-layer model since the middle and high layers are not sensitive to superficial information such as the sentiment task's literal meaning. Similarly, for tasks such as syntactic judgment and semantic reasoning, the accuracy of the low-layer model is much lower than that of the middle-layer or high-layer model, indicating that the low layer does not contain the syntactic knowledge on which these complex tasks rely. Since fusing all layers of information, our model achieves the highest accuracy on all datasets. The above experiments indicate that the model needs to simultaneously learn multiple levels of information, including literal, phrasal, syntactic, semantic, and task information, when selecting task-specific words for a classification task.

Table 4: A case study of the interpretability of different models on three datasets. The top three task-specific words are highlighted and the color saturation indicates the word importance.

Datasets	Models	Texts	Prediction
Subj	BERT	thoughtful even stinging at times and lots of fun.	Subjective
	Vmask	thoughtful even stinging at times and lots of fun.	
	E-VarM	thoughtful even stinging at times and lots of fun .	
AGNews	BERT	athens reuters at the beach volleyball the 2004 olympics is a sell out ...	Sports
	Vmask	athens reuters at the beach volleyball the 2004 olympics is a sell out ...	
	E-VarM	athens reuters at the beach volleyball the 2004 olympics is a sell out ...	
RT	BERT	characterisation has been sacrificed for the sake of spectacle.	Negative
	Vmask	characterisation has been sacrificed for the sake of spectacle.	
	E-VarM	characterisation has been sacrificed for the sake of spectacle.	

Table 5: Bottom (1∼4 layer), middle (5∼8 layer) and top (9∼12 layer) level information impact on model performance.

Datasets	Bottom	Middle	Top	E-VarM
QNLI	86.02	86.95	87.52	**87.94**
QQP	89.31	90.15	90.32	**91.08**
COLA	81.36	81.98	82.35	**83.99**
IMDB	92.34	92.05	91.73	**92.73**
AGNews	92.72	91.18	93.69	**94.00**
Yelp	96.98	96.65	96.39	**97.28**
Subj	96.32	96.35	97.01	**97.30**
RT	87.97	87.21	87.10	**88.67**
SST-1	52.58	50.81	50.31	**53.44**
SST-2	91.54	91.01	90.87	**92.37**

5.4 Visualizing the Text Representations

To present the phenomenon of representations overlap caused by IB and demonstrate that our E-VarM can learn better class discriminative representations, we randomly select 1000 test samples for each dataset and feed them to E-VarM$_{\text{MASK}}$ and E-VarM to obtain text representations. We then visualize these text representations using the T-SNE (Van der Maaten and Hinton, 2008) and show the results for the five datasets in Fig 3 (The results of the comparison with Vmask are in **Appendix B**). As observed, the text representations obtained by E-VarM$_{\text{MASK}}$ have different degrees of overlap, in which the inter-class distance is smaller than that of E-VarM, while the intra-class distance is larger than that of E-VarM. Especially, on the COLA, QNLI, and QQP datasets that involve complex tasks and rely on multi-layer semantics for decisions, there is significant overlap of text representations obtained by E-VarM$_{\text{MASK}}$ among different classes, which would result in indistinguishable classes and reduce the prediction accuracy. In contrast, E-VarM alleviates the problem of inter-class overlap and intra-class dispersion on all datasets through supervised constrastive learning and thus obtains better text representations.

5.5 Visualizing the Interpretation

To further compare the interpretability of the BERT, Vmask, and E-VarM, we conduct case studies on three datasets: Subj, AGNews, and RT. We highlight the top three important words selected by LIME, with the level of color saturation indicating the word's importance. As shown in Table 4, for the same sentences, all three models make correct predictions, it is clear that BERT and Vmask extract many nonsense or task-irrelevant words such as 'at', 'of', and 'out'. In contrast, our model captures more task-specific words, such as 'olympics' related to the topic 'sports', 'characterisation' as the subject of 'sacrificed,' which fits better with the semantics of the input, and 'stinging' and 'fun ', which are more of a subjective expression, showing the outstanding interpretability of our model.

6 Conclusion

In this paper, we propose E-VarM to simultaneously boost the model's interpretability and accuracy. E-VarM combines multi-level information for task-specific words selection, which can adjust the decision basis of the model, and improve the model's interpretability. Further, E-VarM adopts contrastive learning for representation optimization to mitigate the risk of representations overlap, enhancing the model's classification performance. Experimental results on ten benchmark datasets demonstrate the effectiveness of E-VarM.

Acknowledgements

We sincerely thank the reviewers for their insightful comments and valuable suggestions. This work is funded by the National Key Research and Development Program of the Ministry of Science and Technology of China (No. 2021YFB1716201). Thanks for the computing infrastructure provided by Beijing Advanced Innovation Center for Big Data and Brain Computing.

References

Alexander A. Alemi, Ian Fischer, Joshua V. Dillon, and Kevin Murphy. 2017. Deep variational information bottleneck. In *5th International Conference on Learning Representations, ICLR 2017*.

Seo-Jin Bang, Pengtao Xie, Heewook Lee, Wei Wu, and Eric P. Xing. 2021. Explaining A black-box by using A deep variational information bottleneck approach. In *Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Thirty-Third Conference on Innovative Applications of Artificial Intelligence, IAAI 2021, The Eleventh Symposium on Educational Advances in Artificial Intelligence, EAAI 2021, Virtual Event, February 2-9, 2021*, pages 11396–11404. AAAI Press.

Jasmijn Bastings, Wilker Aziz, and Ivan Titov. 2019. Interpretable neural predictions with differentiable binary variables. In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 2963–2977. Association for Computational Linguistics.

Jasmijn Bastings and Katja Filippova. 2020. The elephant in the interpretability room: Why use attention as explanation when we have saliency methods? In *Proceedings of the Third BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP*.

Nicola De Cao, Michael Sejr Schlichtkrull, Wilker Aziz, and Ivan Titov. 2020. 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*, pages 3243–3255. Association for Computational Linguistics.

Hanjie Chen and Yangfeng Ji. 2020. 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. Association for Computational Linguistics.

Jianbo Chen, Le Song, Martin Wainwright, and Michael Jordan. 2018. Learning to explain: An information-theoretic perspective on model interpretation. In *International Conference on Machine Learning*, pages 883–892. PMLR.

George Chrysostomou and Nikolaos Aletras. 2021. Improving the faithfulness of attention-based explanations with task-specific information for text classification. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Gabriel G. Erion, Joseph D. Janizek, Pascal Sturmfels, Scott M. Lundberg, and Su-In Lee. 2019. Learning explainable models using attribution priors. *CoRR*, abs/1906.10670.

Borja Rodríguez Gálvez, Ragnar Thobaben, and Mikael Skoglund. 2020. The convex information bottleneck lagrangian. *Entropy*, 22(1):98.

Robert Geirhos, Patricia Rubisch, Claudio Michaelis, Matthias Bethge, Felix A. Wichmann, and Wieland Brendel. 2019. Imagenet-trained cnns are biased towards texture; increasing shape bias improves accuracy and robustness. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net.

Ziv Goldfeld and Yury Polyanskiy. 2020. The information bottleneck problem and its applications in machine learning. *IEEE Journal on Selected Areas in Information Theory*, 1(1):19–38.

Beliz Gunel, Jingfei Du, Alexis Conneau, and Veselin Stoyanov. 2021. Supervised contrastive learning for pre-trained language model fine-tuning. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net.

Vikram Gupta, Haoyue Shi, Kevin Gimpel, and Mrinmaya Sachan. 2022. Deep clustering of text representations for supervision-free probing of syntax. In *Association for the Advancement of Artificial Intelligence*.

Kishaloy Halder, Alan Akbik, Josip Krapac, and Roland Vollgraf. 2020. Task-aware representation of sentences for generic text classification. In *Proceedings of the 28th International Conference on Computational Linguistics, COLING 2020, Barcelona, Spain (Online), December 8-13, 2020*, pages 3202–3213. International Committee on Computational Linguistics.

John Hewitt and Christopher D Manning. 2019. A structural probe for finding syntax in word representations. In *Proceedings of the 2019 Conference of the North*

American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), pages 4129–4138.

Geoffrey E. Hinton, Oriol Vinyals, and Jeffrey Dean. 2015. Distilling the knowledge in a neural network. *CoRR*, abs/1503.02531.

Alon Jacovi and Yoav Goldberg. 2020. Towards faithfully interpretable NLP systems: How should we define and evaluate faithfulness? In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020*.

Eric Jang, Shixiang Gu, and Ben Poole. 2017. Categorical reparameterization with gumbel-softmax. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net.

Ganesh Jawahar, Benoît Sagot, and Djamé Seddah. 2019. What does BERT learn about the structure of language? In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019*.

Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. 2020. Supervised contrastive learning. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

Taeuk Kim, Jihun Choi, Daniel Edmiston, and Sang-goo Lee. 2020. Are pre-trained language models aware of phrases? simple but strong baselines for grammar induction. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Artemy Kolchinsky, Brendan D. Tracey, and Steven Van Kuyk. 2019. Caveats for information bottleneck in deterministic scenarios. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net.

Ruiqi Li, Xiang Zhao, and Marie-Francine Moens. 2022. A brief overview of universal sentence representation methods: A linguistic view. *ACM Computing Surveys (CSUR)*, 55(3):1–42.

Xianming Li, Zongxi Li, Haoran Xie, and Qing Li. 2021. Merging statistical feature via adaptive gate for improved text classification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 13288–13296.

Bin Liang, Wangda Luo, Xiang Li, Lin Gui, Min Yang, Xiaoqi Yu, and Ruifeng Xu. 2021. Enhancing aspect-based sentiment analysis with supervised contrastive learning. In *CIKM '21: The 30th ACM International Conference on Information and Knowledge Management, Virtual Event, Queensland, Australia, November 1 - 5, 2021*, pages 3242–3247. ACM.

Yi-Shan Lin, Wen-Chuan Lee, and Z. Berkay Celik. 2021. What do you see?: Evaluation of explainable artificial intelligence (XAI) interpretability through neural backdoors. In *KDD '21: The 27th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*.

Xin Lv, Yixin Cao, Lei Hou, Juanzi Li, Zhiyuan Liu, Yichi Zhang, and Zelin Dai. 2021. Is multi-hop reasoning really explainable? towards benchmarking reasoning interpretability. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021*.

Andrew Maas, Raymond E Daly, Peter T Pham, Dan Huang, Andrew Y Ng, and Christopher Potts. 2011. Learning word vectors for sentiment analysis. In *Proceedings of the 49th annual meeting of the association for computational linguistics: Human language technologies*, pages 142–150.

Rabeeh Karimi Mahabadi, Yonatan Belinkov, and James Henderson. 2021. Variational information bottleneck for effective low-resource fine-tuning. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net.

Dong Nguyen. 2018. Comparing automatic and human evaluation of local explanations for text classification. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1069–1078.

Ziqi Pan, Li Niu, Jianfu Zhang, and Liqing Zhang. 2021. Disentangled information bottleneck. In *Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Thirty-Third Conference on Innovative Applications of Artificial Intelligence, IAAI 2021, The Eleventh Symposium on Educational Advances in Artificial Intelligence, EAAI 2021, Virtual Event, February 2-9, 2021*, pages 9285–9293. AAAI Press.

Bo Pang and Lillian Lee. 2005a. Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05)*, pages 115–124, Ann Arbor, Michigan. Association for Computational Linguistics.

Bo Pang and Lillian Lee. 2005b. Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. *arXiv preprint cs/0506075*.

Bhargavi Paranjape, Mandar Joshi, John Thickstun, Hannaneh Hajishirzi, and Luke Zettlemoyer. 2020. An information bottleneck approach for controlling conciseness in rationale extraction. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 1938–1952. Association for Computational Linguistics.

Georgina Peake and Jun Wang. 2018. Explanation mining: Post hoc interpretability of latent factor models for recommendation systems. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD 2018*.

Gregory Plumb, Maruan Al-Shedivat, Ángel Alexander Cabrera, Adam Perer, Eric P. Xing, and Ameet Talwalkar. 2020. Regularizing black-box models for improved interpretability. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020*.

Nazneen Fatema Rajani, Bryan McCann, Caiming Xiong, and Richard Socher. 2019. Explain yourself! leveraging language models for commonsense reasoning. In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 4932–4942. Association for Computational Linguistics.

Danilo Jimenez Rezende and Shakir Mohamed. 2015. Variational inference with normalizing flows. In *Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6-11 July 2015*, volume 37 of *JMLR Workshop and Conference Proceedings*, pages 1530–1538. JMLR.org.

Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. 2016. " 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*, pages 1135–1144.

Joshua David Robinson, Ching-Yao Chuang, Suvrit Sra, and Stefanie Jegelka. 2021. Contrastive learning with hard negative samples. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net.

Timo Schick, Helmut Schmid, and Hinrich Schütze. 2020. Automatically identifying words that can serve as labels for few-shot text classification. In *Proceedings of the 28th International Conference on Computational Linguistics, COLING 2020, Barcelona, Spain (Online), December 8-13, 2020*, pages 5569–5578. International Committee on Computational Linguistics.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D Manning, Andrew Y Ng, and Christopher Potts. 2013. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 conference on empirical methods in natural language processing*, pages 1631–1642.

Xiaofei Sun, Diyi Yang, Xiaoya Li, Tianwei Zhang, Yuxian Meng, Han Qiu, Guoyin Wang, Eduard H. Hovy, and Jiwei Li. 2021. Interpreting deep learning models in natural language processing: A review. *CoRR*, abs/2110.10470.

Zijun Sun, Chun Fan, Qinghong Han, Xiaofei Sun, Yuxian Meng, Fei Wu, and Jiwei Li. 2020. Self-explaining structures improve nlp models. *arXiv preprint arXiv:2012.01786*.

Kyle Swanson, Lili Yu, and Tao Lei. 2020. Rationalizing text matching: Learning sparse alignments via optimal transport. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 5609–5626. Association for Computational Linguistics.

Toshiyuki Tanaka. 1998. A theory of mean field approximation. In *Advances in Neural Information Processing Systems 11, [NIPS Conference, Denver, Colorado, USA, November 30 - December 5, 1998]*, pages 351–360. The MIT Press.

Naftali Tishby, Fernando C. N. Pereira, and William Bialek. 2000. The information bottleneck method. *CoRR*.

Betty Van Aken, Benjamin Winter, Alexander Löser, and Felix A Gers. 2019. How does bert answer questions? a layer-wise analysis of transformer representations. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, pages 1823–1832.

Betty van Aken, Benjamin Winter, Alexander Löser, and Felix A. Gers. 2019. How does BERT answer questions?: A layer-wise analysis of transformer representations. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management, CIKM 2019, Beijing, China, November 3-7, 2019*, pages 1823–1832. ACM.

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

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2018. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 353–355, Brussels, Belgium. Association for Computational Linguistics.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu,

Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. 2019. Huggingface's transformers: State-of-the-art natural language processing. *ArXiv*, abs/1910.03771.

Tailin Wu, Ian Fischer, Isaac L Chuang, and Max Tegmark. 2020. Learnability for the information bottleneck. In *Uncertainty in Artificial Intelligence*, pages 1050–1060. PMLR.

Xing Wu, Chaochen Gao, Liangjun Zang, Jizhong Han, Zhongyuan Wang, and Songlin Hu. 2021. Esimcse: Enhanced sample building method for contrastive learning of unsupervised sentence embedding. *CoRR*, abs/2109.04380.

Lanqing Xue, Xiaopeng Li, and Nevin L. Zhang. 2020. Not all attention is needed: Gated attention network for sequence data. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 6550–6557. AAAI Press.

Dejiao Zhang, Shang-Wen Li, Wei Xiao, Henghui Zhu, Ramesh Nallapati, Andrew O. Arnold, and Bing Xiang. 2021. Pairwise supervised contrastive learning of sentence representations. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021*, pages 5786–5798. Association for Computational Linguistics.

Xiang Zhang, Junbo Zhao, and Yann LeCun. 2015. Character-level convolutional networks for text classification. *Advances in neural information processing systems*, 28.

A Datasets

This paper uses ten widely studied datasets that cover a wide range of application domains, which are described in detail below:

For sentiment analysis, we use five benchmark datasets in which **RT** [3] (Pang and Lee, 2005a), **IMDB** [4] (Maas et al., 2011), **SST-2** [5] (Socher et al., 2013) and **Yelp (YelpReviewPolarity)** [6] (Zhang et al., 2015) are four binary sentiment polarity datasets with each sentence annotated as positive or negative. And **SST-1** [7] (Socher et al., 2013) is a fine-grained sentiment dataset derived from Stanford Sentiment Treebank with five balanced labels (negative, somewhat negative, neutral, somewhat positive, positive).

For topic categorization, we use **AG's News**[8] (Zhang et al., 2015) dataset where each article only has a title and description and can be categorized into one of the four main classes: "World", "Sports", "Business", and "Technology".

For grammatical judgment, we adopt **COLA** [9] (Wang et al., 2018) dataset that published by New York University with each sentence marked whether there are grammatical errors or not.

For semantic inference, we employe **QQP** [10] (Wang et al., 2018) dataset to determine whether a questions pair is semantically equivalent, and the **QNLI** [11] (Wang et al., 2018) dataset to judge a question-sentence pair is entailment relation or not.

Additionally, we leverage **Subj** [12] (Pang and Lee, 2005b) dataset to carry out subjective / objective classification, which contains 5000 subjective and 5000 objective sentences, respectively.

B Visualization Supplement

We visualize the text representations for the remaining five datasets . As shown in Fig. 4, the text representations obtained by E-VarM have a larger inter-class distance and a smaller intra-class distance than that obtained by Vmask. This phenomenon indicates that our model can adjust the text representation and alleviate the problem of representations overlap.

[3] https://www.cs.cornell.edu/people/pabo/movie-review-data/rt-polaritydata.tar.gz

[4] http://ai.stanford.edu/ amaas/data/sentiment/aclImdb_v1.tar.gz

[5] https://drive.google.com/uc?export=download&id=0Bz8a_Dbh9QhbNUpYQ2N3SGlFaDg

[6] https://drive.google.com/uc?export=download&id=0Bz8a_Dbh9QhbNUpYQ2N3SGlFaDg

[7] https://drive.google.com/uc?export=download&id=0Bz8a_Dbh9QhbNUpYQ2N3SGlFaDg

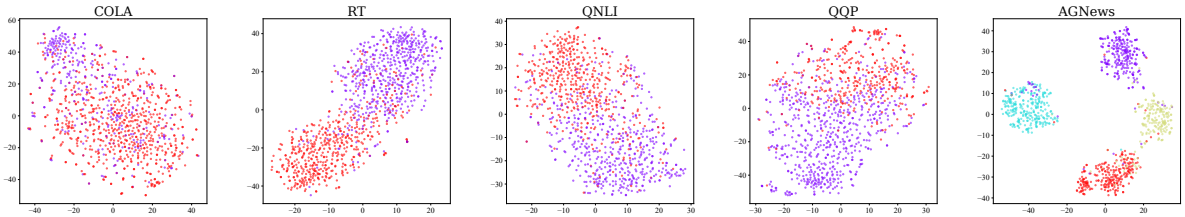
[8] http://groups.di.unipi.it/ gulli/AG_corpus_of_news_articles.html

[9] https://gluebenchmark.com/tasks

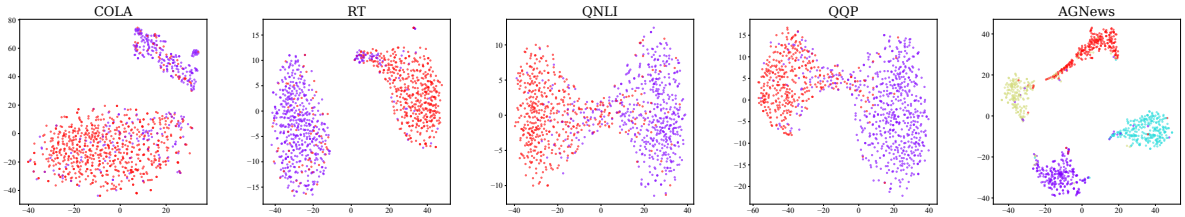
[10] https://gluebenchmark.com/tasks

[11] https://gluebenchmark.com/tasks

[12] https://www.cs.cornell.edu/people/pabo/movie-review-data/



(a) Vmask



(b) E-VarM

Figure 4: T-SNE visualization comparison. The upper is Vmask, and the lower is our method.