Structural Entropy Guided Probabilistic Coding

Xiang Huang[1], Hao Peng[1,2*], Li Sun[3], Hui Lin[4], Chunyang Liu[5], Jiang Cao[6], Philip S. Yu[7]

[1]Beihang University
[2]Guangdong Laboratory of Artificial Intelligence and Digital Economy
[3]North China Electric Power University
[4]China Academic of Electronics and Information Technology
[5]Didi Chuxing
[6]Academy of Military Sciences
[7]University of Illinois Chicago
{huang.xiang, penghao}@buaa.edu.cn, ccesunli@ncepu.edu.cn, linhui@cetc.com.cn
liuchunyang@didiglobal.com, amscaojiang@126.com, psyu@uic.edu

Abstract

Probabilistic embeddings have several advantages over deterministic embeddings as they map each data point to a distribution, which better describes the uncertainty and complexity of data. Many works focus on adjusting the distribution constraint under the Information Bottleneck (IB) principle to enhance representation learning. However, these proposed regularization terms only consider the constraint of each latent variable, omitting the structural information between latent variables. In this paper, we propose a novel structural entropy-guided probabilistic coding model, named SEPC. Specifically, we incorporate the relationship between latent variables into the optimization by proposing a structural entropy regularization loss. Besides, as traditional structural information theory is not well-suited for regression tasks, we propose a probabilistic encoding tree, transferring regression tasks to classification tasks while diminishing the influence of the transformation. Experimental results across 12 natural language understanding tasks, including both classification and regression tasks, demonstrate the superior performance of SEPC compared to other state-of-the-art models in terms of effectiveness, generalization capability, and robustness to label noise.

Code — https://github.com/SELGroup/SEPC

Introduction

Probabilistic embedding (Vilnis and McCallum 2015) is a flexible representation learning method aiming to learn the underlying probability distribution of data. It has been broadly applied to various domains such as graph structural learning (Sun et al. 2022), computer vision (Kim et al. 2021; Oh et al. 2019; Shi and Jain 2019; Fischer 2020), and natural language processing (Mahabadi, Belinkov, and Henderson 2021; Hu et al. 2024, 2022). In contrast to deterministic embedding (Dong, Yan, and Wang 2024; Xu et al. 2024), which maps the input into a fixed latent variable representation, probabilistic embedding represents each data point as a probability distribution. Hence, probabilistic embedding inherently accounts for the uncertainty and complexity of data by controlling the spread of the probability density over the learning latent space (Oh et al. 2019), showcasing better discriminative ability and robustness.

The mainstream probabilistic embedding methods are grounded in the Information Bottleneck (IB) principle (Tishby, Pereira, and Bialek 2000; Tishby and Zaslavsky 2015). IB aims to find compressed representations that maintain as much information as possible for the prediction task while removing as much irrelevant information as possible. Specifically, it seeks the latent representation Z that is maximally informative about the target Y (i.e., maximize mutual information $I(Y;Z)$) while being minimally informative about the input data X (i.e., minimize mutual information $I(X;Z)$) (Sun et al. 2022). The former target is typically achieved with common task losses like cross entropy (CE) loss or mean squared error (MSE) loss, whereas various regularization losses are proposed for the latter goal. VIB (Alemi et al. 2017) assumes the prior distribution of Z is the standard normal distribution and utilizes Kullback–Leibler (KL) divergence to regularize the learning distribution $p(z|x)$. Sparse IB (Chalk, Marre, and Tkacik 2016) changes the prior distribution of VIB to the Student-t distribution to achieve relevant and sparse coding. MEIB (An, Jammalamadaka, and Chong 2023) lifts the prior distribution constraint of VIB and instead uses maximum conditional entropy $H(Z|X)$ as the only regularization. SPC (Hu et al. 2024) omits the decoder of VIB and proposes an additional structured regularization that encourages class-level uniformity within the latent space under the multivariate Gaussian distribution. However, all of them focus solely on the individual latent variable Z or the constraint of Z with the label Y, neglecting the structural information between latent variables.

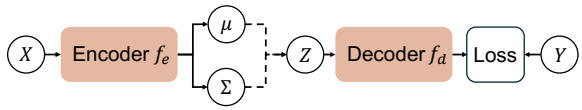
In recent years, structural entropy theory (Li and Pan 2016) has demonstrated its advantage in capturing hierarchical structural information and has been widely used in various fields like node classification (Duan et al. 2024), graph structural learning (Zou et al. 2023), and contrastive learning (Wu et al. 2023). It considers the structural infor-

*Corresponding author.

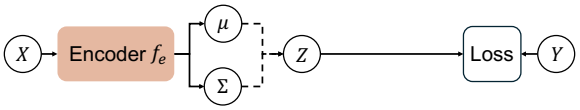
mation of the original inputs by modeling the input data as a graph and then converting the graph into an encoding tree. The data points are the leaf nodes of the encoding tree, and each upper node represents a partition, resulting in a hierarchical clustering of the input data. Low-depth tree nodes depict more coarse-grained clusters of the input data. Each node in the encoding tree has its own structural entropy. The structural entropy of the encoding tree is calculated by summing the structural entropy of all non-root nodes, representing the overall structural information of the input. Previous work (Wang et al. 2023; Zeng, Peng, and Li 2023) mostly focuses on minimizing the structural entropy of the encoding trees to obtain the optimized encoding tree or embeddings of input data, aiming at learning as much task-related information as possible. However, the potential for using structural entropy for regularization remains underexplored. Additionally, as the structural entropy is designed for classification tasks, how to effectively leverage it in the regression task is still a problem.

In this paper, we propose a structural entropy-guided probabilistic coding model, named SEPC. We present a structural entropy-based regularization loss that incorporates structural information between latent variables. Specifically, we first construct the adjacency matrix based on the similarity between embeddings of latent variables and propose to maximize the structural entropy of the induced graph, which helps improve the generalization of the model by separating the probabilistic distribution of each latent variable. Additionally, we design a probabilistic encoding tree to adapt our structural entropy loss in regression tasks. We first discretize and soften regression labels into soft classification labels (i.e., each data point belongs to multiple classes with varying probabilities), diminishing the influence of unsuitable classification caused by using only discretization (Pintea et al. 2023). To adapt structural entropy to such soft labels, we relax the constraint that one child belongs to one parent in the encoding tree, allowing each child to connect to all upper-level nodes with varying probabilities. Extensive experiments are conducted on 12 natural language understanding tasks, including 10 classification tasks and 2 regression tasks. Comparative results and analysis demonstrate that the proposed SEPC enjoys superior effectiveness, generalization, and robustness compared to the state-of-the-art (SOTA) baselines. The main contributions are summarized as follows:

- We present a structural entropy based regularization loss, incorporating the structural information between data points into model regularization. To our knowledge, this is the first time that maximizing structural entropy has been utilized as a regularization loss.
- We propose a probabilistic encoding tree for soft classification labels and present an effective method to utilize structural entropy for regression tasks for the first time.
- Extensive experiments on 12 datasets demonstrate that SEPC achieves SOTA performance in classification and regression tasks regarding effectiveness, generalization, and robustness.



(a) Encoder-Decoder architecture.



(b) Encoder-only architecture.

Figure 1: Two common architectures of probabilistic coding.

Preliminaries

In this section, we present the basic concepts of probabilistic coding, the encoding tree, and structural entropy.

Probabilistic Coding

The classical probabilistic coding model employs an encoder-decoder architecture, as shown in Figure 1(a). The encoder f_e maps input $x \in X$ to a Gaussian distribution $\mathcal{N}(z; \mu, \Sigma)$. All distributions of z consist of the embedding space of the latent variable Z. The re-parameterization trick (Kingma and Welling 2013) is then used to sample z from the distribution while keeping the gradient unbiased. Finally, z is mapped by the decoder to $f_d(z)$ to predict the label $y \in Y$. The work (Hu et al. 2024) also proposes an encoder-only architecture for probabilistic coding (as shown in Figure 1(b)), omitting the decoder and directly predicting y using the sample from the learned distribution.

Under the Markov chain constraint $Y \rightarrow X \rightarrow Z$, the probabilistic coding follows the Information Bottleneck principle and aims to learn the minimal sufficient information for representation Z:

$$Z = \underset{Z}{argmin} -I(Z;Y) + \beta I(Z;X), \tag{1}$$

where $I(Z;Y)$ is the mutual information between Z and Y, $I(Z;X)$ is the mutual information between Z and X, and β is the Lagrangian multiplier trading off sufficiency and minimality. Assuming $z \in Z$ follows the Gaussian distribution, the objective of probabilistic coding is as follows:

$$\mathcal{L}_{PC} = \mathbb{E}_{z\sim p(z|x)} \left[-\log q(y|z)\right] + \beta \mathrm{KL}\left[p(z|x), r(Z)\right]. \tag{2}$$

Here, KL refers to the KL divergence operator, $p(z|x) = \mathcal{N}(z; \mu, \Sigma)$ is learned by the encoder f_e, $r(Z)$ is the expected prior distribution, and $r(Z) = \mathcal{N}(z; \mathbf{0}, \mathbf{I})$ in general. $q(y|z)$ is the variational approximation to $p(y|z)$ and is calculated by the decoder f_d or by a non-parametric operator like the softmax function in the encoder-only architecture (Hu et al. 2024).

Encoding Tree

Given a graph $G = \{X, E, W\}$, X is the set of input data points, E is the edge set, and $W \in \mathbb{R}^+$ is the edge weight set. For each point $x \in X$, its degree d_x is defined as the sum of the weights of edges associated with it. The encoding tree $\mathcal{T}$ of G is a multi-child tree with the following properties:

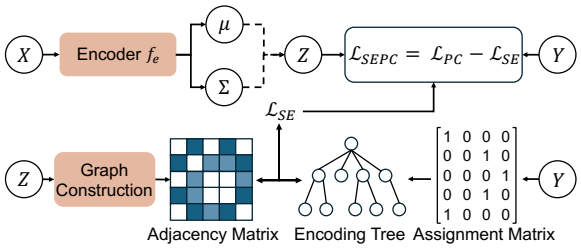


Figure 2: The overall model of SEPC.

(1) Each tree node α corresponds to a subset of data points $T_\alpha \subseteq X$. Especially, for the root node λ of $\mathcal{T}$, we define the points set it associated with as $T_\lambda = X$. For the leaf node α at the last depth, T_α is a singleton containing a single data point $x \in X$. If the leaf node α is not at the last depth, T_α is $\emptyset$. (2) For each non-leaf tree node α, its i-th immediate child is $\alpha^{<i>}$, and its parent node is denoted as α^-. (3) For each non-leaf tree node α, $T_\alpha = \bigcup_{i=1}^{N_\alpha} T_{\alpha^{<i>}}$, N_α is the number of children of α. With these properties, each depth of a node in the encoding tree depicts a partition of the data point set X, and lower depth means a more coarse-grained partition.

Structural Entropy

The structural entropy is defined under the graph G and the encoding tree $\mathcal{T}$ as follows:

$$H^{\mathcal{T}}(G) = \sum_{\alpha \in \mathcal{T}, \alpha \neq \lambda} H^{\mathcal{T}}(G; \alpha), \tag{3}$$

$$H^{\mathcal{T}}(G; \alpha) = -\frac{g_\alpha}{\mathrm{vol}(G)} \log_2 \frac{\mathcal{V}_\alpha}{\mathcal{V}_{\alpha^-}}. \tag{4}$$

Here, g_α is the sum of the weights of the edges that connect points inside T_α with points outside T_α (i.e., the weights of the cut edges between T_α and its complement set $T_\alpha^{\complement}$). The volume of G, denoted as $\mathrm{vol}(G)$, is the sum of the degrees of all data points X, i.e., $\mathrm{vol}(G) = \sum_{x \in X} d_x$. $\mathcal{V}_\alpha = \sum_{x \in T_\alpha} d_x$ is the volume of T_α, and α^- is the parent node of α.

Proposed Method

In this section, we elaborate on the proposed structural entropy based regularization loss of SEPC, introduce the probabilistic encoding tree for soft classification labels, and describe how to utilize it in regression tasks. We adopt the encoder-only architecture (Hu et al. 2024) for probabilistic coding, and the overall model of SEPC is shown in Figure 2.

Structural Entropy based Regularization Loss

Previous works only consider the individual latent variable in the regularization loss, ignoring the structural information between latent variables. To capture the structural information, we incorporate structural entropy into the regularization loss, as it inherently considers the self-organization of data. As illustrated in Figure 2, the input data X is first encoded into the probabilistic embedding H_Z. The graph G is constructed from Z as follows:

$$A = \sigma(H_Z \times H_Z^T), \tag{5}$$

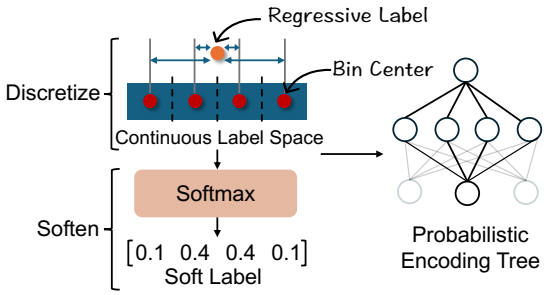


Figure 3: Probabilistic encoding tree for regression tasks.

where H_Z is the embedding of Z, and σ is the sigmoid activation function to ensure positive values for the adjacency matrix A.

The construction of the encoding tree is also straightforward. We treat the labels as the optimal partition for the data and construct a three-tier encoding tree. The nodes in the intermediate layer represent the classes of the classification task, and each leaf node (i.e., the input data X) is assigned to an intermediate node according to its label. We define an assignment matrix $C \in \{0,1\}^{n \times r}$, where n is the number of leaf nodes and r is the number of intermediate nodes. $C_{ij} = 1$ means the i-th leaf node belongs to the j-th class. To enhance the capability of the latent representations, we propose maximizing the structural entropy of the intermediate layer nodes, constraining the probabilistic distribution of the latent variables to ensure separation. The structural entropy of the intermediate layer nodes for the three-tier encoding tree is as follows:

$$H_C^{\mathcal{T}}(G) = \sum_{j=1}^{r} \frac{-g_{\alpha_j}}{\mathrm{vol}(G)} \log_2 \frac{\mathcal{V}_{\alpha_j}}{\mathrm{vol}(G)}, \tag{6}$$

where r is the number of classes, g_{α_j} is the sum of the weights of the cut edges between T_{α_j} and its complement set $T_{\alpha_j}^{\complement}$, $\mathcal{V}_{\alpha_j}$ is the volume of T_{α_j}, and $\{\alpha_1, \ldots, \alpha_r\}$ is the intermediate layer nodes in the encoding tree. Utilizing the adjacency matrix A and the assignment matrix C, the regularization loss format of $H_C^{\mathcal{T}}(G)$ is as follows:

$$\mathcal{L}_{SE} = -\sum_{j=1}^{r} \frac{\left((\mathbf{1} - C)^T A C\right)_{jj}}{\mathrm{sum}(A)} \times \log_2 \frac{\left(\mathbf{1}^T A C\right)_{jj}}{\mathrm{sum(A)}}. \tag{7}$$

Here, $\mathbf{1}$ is the full-one matrix with shape $n \times r$, the operator $\mathrm{sum}(\cdot)$ sums up the matrix to a scalar, and $(\cdot)_{jj}$ selects the value in the j-th row and the j-th column of the matrix. The overall loss of SEPC is as follows:

$$\mathcal{L}_{SEPC} = \mathcal{L}_{PC} - \gamma \mathcal{L}_{SE}, \tag{8}$$

where γ is a hyperparameter controlling the weight of our structural entropy based regularization loss $\mathcal{L}_{SE}$.

Probabilistic Encoding Tree for Regression Tasks

Discretization is a widely used method to transform a regression task into a classification task by binning continuous labels into discrete classes (Muthukumar et al. 2021; Stewart et al. 2023). However, as the binning borders need to

be predefined, inappropriate borders can lead to unbalanced or indistinguishable classification labels, hampering model performance (Pintea et al. 2023). Softening labels mitigates this issue (Ma et al. 2023), as it allows each data point to belong to all classes with different probabilities to express tendencies. We propose a probabilistic encoding tree to utilize structural entropy theory in such soft classification labels. It loosens the constraint that one child node is only assigned to one parent node, allowing the child node to connect with all up-depth nodes with different probabilities.

As shown in Figure 3, during the discretized period, we first bin the entire regression label value space into r classes. Then, we calculate the distance between the regressive label Y and the centers $P = \{P_1, \ldots, P_r\}$ of the r bins:

$$D = |Y^T - P|, \tag{9}$$

where $D \in \mathbb{R}^{n \times r}$, n is the number of data points, and the i-th row of D denotes the distance between the i-th data point and the r bin centers. The soft label is then calculated during the softening period as follows:

$$Y' = \mathrm{softmax}(-D), \tag{10}$$

where $-D$ ensures that a closer distance to the bin center corresponds to a higher probability of belonging to this class. The structural entropy of the intermediate layer nodes for the three-tier probabilistic encoding tree $H_C^{\mathcal{T}}(G)$ is then defined as follows:

$$\mathcal{V}'_{\alpha_j} = \sum_{x_i \in X} Y'_{ij} d_{x_i}, \tag{11}$$

$$H^{\mathcal{T}}(G; \alpha_j) = -\frac{g'_{\alpha_j}}{\mathrm{vol}(G)} \log_2 \frac{\mathcal{V}'_{\alpha_j}}{\mathrm{vol}(G)}, \tag{12}$$

$$H_C^{\mathcal{T}}(G) = \sum_{j=1}^{r} H^{\mathcal{T}}(G; \alpha_j). \tag{13}$$

Here, Y'_{ij} denotes the soft label of the i-th data point x_i regarding to the j-th class, and d_{x_i} is the degree of x_i. α_j represents the j-th intermediate layer nodes in the probabilistic encoding tree. For g'_{α_j}, the weight of cut edges should be multiplied by the probability of one vertex belonging to T_{α_j} and the other belonging to $T^{\complement}_{\alpha_j}$. Letting the assignment matrix $C = Y'$, the structural entropy loss for the probabilistic encoding tree is as follows:

$$\mathcal{L}_{SE} = -\sum_{j=1}^{r} \frac{\left((\mathbf{1} - C)^T A C\right)_{jj}}{\mathrm{sum}(A)} \times \log_2 \frac{\left(\mathbf{1}^T A C\right)_{jj}}{\mathrm{sum(A)}}. \tag{14}$$

It is equivalent to Equation 7 in the formula, except that the elements of the assignment matrix C are probabilities between 0 and 1. Thus far, we have presented an effective method to utilize structural entropy for regression tasks.

Experiments

In this section, we conduct extensive experiments to evaluate the effectiveness, generalization capability, and robustness of SEPC. For fairness, all results are reported as the average and standard deviation of metrics tested with five random seeds, as in other works.

Experiment Setups

Datasets Following Hu et al. (2024), we evaluate SEPC on 10 classification task datasets and 2 regression task datasets. For classification tasks, 7 datasets about tweet semantic analysis are used: Emoji (Barbieri et al. 2018), Emotion (Mohammad et al. 2018), Hate (Basile et al. 2019), Irony (Van Hee, Lefever, and Hoste 2018), Offensive (Zampieri et al. 2019), Sentiment (Rosenthal, Farra, and Nakov 2017), and Stance (Mohammad et al. 2016). Additionally, we also experiment on three emotion-related datasets from different domains: ISEAR (Scherer and Wallbott 1994), MELD (Poria et al. 2019), and GoEmotions (Demszky et al. 2020). For regression tasks, we utilize STS-B (Cer et al. 2017) and Claire (Roth, Anthonio, and Sauer 2022) for evaluation.

Evaluation Metric We use the same metric as in previous works. The macro-averaged F1 score across all classes is reported for most classification datasets. Following Hu et al. (2024), we report the macro-averaged F1 score of favor and against classes for the Stance dataset, the F1 score of the ironic class for the Irony dataset, and the macro-averaged recall for the Sentiment dataset. For regression tasks, we report both Pearson and Spearman correlation coefficients.

Baselines We compare SEPC with two categories of classic baselines: universal models and fine-tuned representation models. The baseline results are collected from the work of Hu et al. or evaluated using the source code provided by the authors. In the universal models, we compare with SVM (Cortes and Vapnik 1995), FastText (Joulin et al. 2017), BiLSTM (Hochreiter and Schmidhuber 1997), and GPT-3.5[1]. For the fine-tuned models, we use bert-base-uncased (Devlin et al. 2019) and roberta-base (Liu et al. 2020) as the backbone and fine-tune them on the evaluation datasets. We compare with four deterministic embedding baselines: cross-entropy (CE) for classification tasks and mean squared error (MSE) for regression tasks, CE+CP (Pereyra et al. 2017), CE/MSE+AT (Miyato, Dai, and Goodfellow 2017), and CE+SCL (Gunel et al. 2021). Besides, we compare SEPC with four probabilistic embedding models: VIB (Alemi et al. 2017), MINE-IB (Belghazi et al. 2018), MEIB (An, Jammalamadaka, and Chong 2023), and SPC (Hu et al. 2024).

Parameter Settings The training epoch number is 20, and the maximum patience for early stopping is 5 epochs. The learning rate is 5e-5 in all datasets. A linear learning rate warm-up is applied over the first 10% of the training data. The batch size is uniformly set to 128. The trade-off parameter β and the weight parameter γ are searched from $\{1e-2, 1e-1, 1, 10\}$. We set the class number $r = 5$ for the STS-B dataset and $r = 4$ for the Claire dataset, as their labels range from 0–5 and 1–5, respectively. All experiments are conducted on two NVIDIA RTX A6000 GPUs.

Evaluations

Classification Tasks We conduct comparative experiments with the baselines on 10 classification datasets and

[1]https://openai.com/index/chatgpt/

Method	Emoji	Emotion	Hate	Irony	Offensive	Sentiment	Stance	ISEAR	MELD	GoEmotions	Avg.
SVM	29.30	64.70	36.70	61.70	52.30	62.90	67.30	-	-	-	-
FastText	25.80	65.20	50.60	63.10	73.40	62.90	65.40	-	-	-	-
BiLSTM	24.70	66.00	52.60	62.80	71.70	58.30	59.40	-	-	-	-
GPT-3.5	6.34±0.01	73.23±0.18	48.30±0.11	66.81±3.26	63.71±0.13	40.40±3.13	39.45±0.10	67.22±0.09	41.46±0.11	25.21±0.08	47.21
BERT backbone											
CE	22.30±0.60	76.05±1.41	44.67±1.78	59.38±3.01	80.16±1.26	70.54±0.44	65.21±0.71	67.17±0.78	39.80±0.84	46.29±0.79	57.16
CE+CP	21.91±0.71	76.28±1.20	45.97±2.93	64.06±2.41	78.99±1.57	70.68±0.31	65.83±0.39	67.20±0.95	39.54±1.61	46.39±0.63	57.69
CE+AT	22.93±0.70	75.08±1.23	46.30±3.61	64.23±2.04	79.68±1.59	70.55±0.57	66.46±1.13	65.70±0.69	39.84±0.38	47.37±0.54	57.81
CE+SCL	21.72±0.51	75.43±1.37	45.86±1.15	65.39±2.46	80.20±0.56	70.70±0.79	65.34±0.60	67.54±0.64	40.00±1.96	46.50±0.46	57.87
VIB	21.31±0.62	77.37±0.71	45.99±1.93	63.82±1.00	80.37±1.11	70.39±0.31	65.43±0.60	67.24±0.57	38.52±0.51	45.89±1.10	57.63
MINE-IB	21.29±0.31	76.60±0.41	47.64±2.11	65.86±2.57	78.67±2.28	69.85±0.54	65.35±0.88	67.62±0.40	41.23±0.67	46.87±0.42	58.10
MEIB	21.87±0.73	76.70±0.82	48.27±1.72	65.87±2.14	80.49±0.81	70.55±0.57	65.59±1.58	67.44±0.50	39.30±0.61	46.26±0.81	58.23
SPC	24.19±1.55	77.15±0.73	57.48±2.99	65.85±1.07	80.65±0.78	70.74±0.12	67.17±1.08	68.94±0.35	42.68±0.94	47.62±1.38	60.25
SEPC	24.85±0.31	78.58±0.25	62.44±2.08	69.56±1.14	82.14±0.59	71.35±0.21	69.25±0.78	69.77±0.26	43.23±0.71	51.16±0.35	62.23
- w/o SE	22.49±0.43	76.63±1.07	56.54±0.68	67.10±0.54	80.31±0.79	70.57±0.54	66.84±0.83	68.69±0.14	42.31±0.39	46.68±0.32	59.82
RoBERTa backbone											
CE	30.25±1.32	77.41±1.33	45.49±4.70	57.99±4.96	78.74±2.20	71.80±0.93	66.78±1.34	70.00±0.45	39.23±0.41	46.64±1.15	58.43
CE+CP	31.12±0.84	77.54±0.70	48.59±3.28	58.75±6.19	79.50±0.98	72.82±0.29	66.89±1.67	70.58±0.71	40.74±0.89	47.98±0.65	59.45
CE+AT	32.00±0.93	77.30±1.07	44.71±4.76	60.17±3.17	79.81±1.11	72.51±0.44	67.81±0.95	70.97±0.68	40.10±0.60	47.89±1.21	59.33
CE+SCL	31.09±1.85	76.98±2.02	49.51±2.86	60.71±4.23	80.39±0.83	73.16±0.44	66.73±1.54	70.26±0.45	40.64±1.02	47.87±0.86	59.72
VIB	29.71±0.79	77.99±0.86	49.39±3.08	59.93±4.57	79.63±0.66	72.81±0.39	68.40±0.52	70.74±0.44	38.94±0.55	46.23±0.18	59.38
MINE-IB	31.70±0.45	78.79±0.58	46.39±2.82	57.39±8.27	79.76±0.67	72.85±0.56	67.27±1.00	70.15±0.58	41.80±2.14	48.88±1.04	59.50
MEIB	29.94±1.30	78.73±0.90	49.34±2.42	60.54±2.70	79.68±0.98	72.78±0.29	67.89±1.70	70.86±0.61	39.00±0.37	47.18±1.15	59.59
SPC	32.54±0.48	79.01±0.61	59.80±1.32	65.31±1.91	80.98±1.36	72.96±0.22	69.02±0.63	71.01±0.59	43.99±0.29	50.04±0.60	62.47
SEPC	**32.90±0.22**	**79.82±0.54**	**63.41±1.27**	**70.02±1.22**	**82.09±0.46**	**73.18±0.34**	**70.33±0.53**	**71.92±0.19**	**44.64±0.42**	**51.55±0.83**	**63.99**
- w/o SE	31.05±0.63	79.25±0.33	57.13±5.10	67.20±0.86	80.74±0.83	72.73±0.19	69.06±0.41	71.11±0.92	43.23±1.03	48.52±0.86	62.00

Table 1: Classification evaluation (%) results. The best results are bolded. w/o SE refers to SEPC without the proposed structural entropy based regularization loss.

Method	STS-B		Claire		Avg.
	Spearman	Pearson	Spearman	Pearson	
MSE	88.33±0.32	88.80±0.36	50.37±5.90	49.10±5.74	69.15
MSE+AT	88.40±0.50	89.01±0.37	53.09±0.64	51.87±0.65	70.59
VIB	88.45±0.50	89.01±0.40	52.86±0.88	51.66±0.78	70.49
MEIB	88.61±0.14	89.13±0.17	52.85±0.72	51.39±0.81	70.50
SPC	88.71±0.19	89.31±0.24	53.11±0.95	52.21±0.81	70.84
SEPC	**89.10±0.29**	**89.64±0.20**	**54.66±0.69**	**53.81±0.84**	**71.80**
- w/o soft	88.90±0.29	89.27±0.31	53.65±0.64	52.85±0.44	71.17

Table 2: Regression evaluation (%) results with the RoBERTa backbone. The best results are bolded, and the second-best results are underlined. w/o soft refers to SEPC without the soft label and probabilistic encoding tree.

report the results in Table 1. Both on the BERT backbone and the RoBERTa backbone, SEPC outperforms all other baselines with 2.02%-5.07% and 1.52%-5.56% average metric improvements, respectively. Compared to SPC, which is also an encoder-only architecture-based probability coding model, SEPC still shows superior performance across all datasets. These experimental results demonstrate the effectiveness of our proposed structural entropy based regularization loss $\mathcal{L}_{SE}$. The most notable enhancement occurs in the Hate and the Irony dataset, where SEPC with the RoBERTa backbone surpasses all baselines with improvements ranging from 3.61% to 26.71% and 4.71% to 12.63%, respectively. As the Hate datasets exhibit topic imbalance between the train and test sets, and the Irony dataset has higher requirements on language understanding because the semantics of ironic text are subtle compared to non-ironic text, the superior performance also demonstrates the better generalization capability of SEPC.

We also conduct an ablation study on SEPC w/o SE model, disable the proposed $\mathcal{L}_{SE}$, and report the results in Table 1. It is noteworthy that, despite SEPC w/o SE model being the same as SPC w/o S model (Hu et al. 2024), we report our experimental results as the hyperparameters and experiment environments differ. The absence of $\mathcal{L}_{SE}$ leads to a performance decrease of an average of 2.41% and 1.99% on the BERT and RoBERTa backbones, respectively. This indicates the effectiveness of our proposed structural entropy based regularization loss.

Regression Tasks We experiment with regression tasks on STS-B and Claire datasets and report Spearman and Pearson correlation coefficients results in Table 2. All methods use RoBERTa as the backbone. SEPC outperforms all other baselines across all datasets, with an average of 0.96%-2.65% performance improvement. This demonstrates the effectiveness of SEPC on the regression tasks. To better understand our proposed probabilistic encoding tree, we conduct an ablation study by removing the soft label and probabilistic encoding tree. Instead, we assign each sample to the class with the closest distance to the class bin center and use the normal encoding tree to calculate $\mathcal{L}_{SE}$. As shown in Table 2, SEPC outperforms SEPC without the soft label and probabilistic encoding tree. This proves the information loss of directly discretizing regression labels and also indicates the effectiveness of our proposed method of softening and probabilistic encoding in the regression tasks.

Robustness Analysis To evaluate the robustness of SEPC, we introduce noise by randomly flipping 10%, 20%, and 30% of the labels in the training datasets to any class with the same probability. The experimental results are reported in

Method	Noisy	Emoji	Emotion	Hate	Irony	Offensive	Sentiment	Stance	ISEAR	MELD	GoEmotions	Avg.
CE	10%	30.66±0.89	78.15±0.88	47.06±5.40	56.90±4.58	79.46±0.80	72.36±0.74	67.39±1.86	70.40±0.97	42.01±1.94	47.85±1.08	59.22
VIB	10%	30.74±0.48	77.78±2.05	47.64±1.57	58.66±10.60	79.96±0.73	72.13±0.54	67.54±1.20	70.85±0.33	38.63±0.89	47.30±1.65	59.12
MINE-IB	10%	31.14±0.65	78.04±1.03	47.19±3.29	56.80±8.63	78.36±1.46	72.42±0.47	67.16±1.51	70.34±0.44	42.32±1.65	48.56±1.41	59.23
MEIB	10%	31.02±0.47	78.94±0.46	49.28±4.58	57.21±8.07	80.19±0.83	72.09±0.68	68.26±0.68	70.85±0.38	38.67±0.97	46.93±1.06	59.34
SPC	10%	32.25±0.69	78.88±0.47	56.13±5.36	58.88±4.94	80.14±0.28	72.76±0.06	68.57±1.01	71.10±0.62	43.90±1.13	49.32±1.22	61.19
SEPC	10%	**32.92±0.39**	**79.17±0.56**	**60.93±1.96**	**69.86±1.33**	**81.33±0.21**	**72.99±0.18**	**69.33±0.99**	**71.61±0.35**	**44.57±0.19**	**51.53±0.64**	**63.42**
CE	20%	31.96±0.88	77.01±1.51	49.12±0.72	60.82±3.56	79.54±1.64	72.06±0.63	68.49±1.20	70.32±0.26	40.16±1.94	47.78±0.84	59.73
VIB	20%	30.46±0.59	79.00±0.49	47.91±2.20	60.67±4.82	79.15±1.22	72.26±0.29	66.83±0.52	71.02±0.25	39.33±1.47	47.83±1.38	59.45
MINE-IB	20%	30.31±0.97	77.84±0.98	46.23±3.23	57.43±8.41	78.65±0.91	72.02±0.83	66.83±1.82	69.26±0.52	42.31±1.58	47.55±0.99	58.84
MEIB	20%	30.84±0.75	78.38±0.88	50.02±5.18	55.12±7.07	78.17±2.55	71.63±1.11	68.05±0.81	70.68±0.38	39.09±0.87	47.29±1.22	58.93
SPC	20%	32.51±0.83	77.97±1.12	55.41±6.00	66.40±4.26	80.33±0.48	72.50±0.55	68.89±1.60	71.10±0.39	43.96±0.50	49.04±0.42	61.81
SEPC	20%	**33.04±0.19**	**79.55±0.42**	**60.30±1.80**	**69.61±1.51**	**81.66±0.44**	**72.97±0.30**	**69.81±0.64**	**71.68±0.26**	**44.50±0.86**	**51.64±0.42**	**63.48**
CE	30%	31.82±0.75	77.61±0.90	50.69±2.80	58.90±11.45	78.11±2.07	70.15±0.50	69.07±1.07	70.74±0.56	40.61±2.06	47.76±2.29	59.55
VIB	30%	30.85±0.53	78.23±0.79	48.22±1.97	58.81±8.84	79.38±0.62	72.15±0.52	67.59±0.93	70.27±0.74	38.71±1.19	47.16±1.32	59.14
MINE-IB	30%	30.12±0.79	77.82±1.24	46.05±3.94	56.02±7.24	78.26±1.58	72.23±0.74	65.56±2.67	69.55±0.92	39.46±1.82	46.71±1.87	58.18
MEIB	30%	30.74±0.87	77.99±0.69	49.98±4.00	57.57±5.19	72.53±5.53	71.83±0.40	67.88±0.68	69.86±1.24	39.39±1.06	47.43±1.52	58.52
SPC	30%	32.27±0.48	78.13±1.13	56.04±7.44	59.27±8.56	80.32±0.53	72.44±0.36	69.77±0.93	70.91±0.30	43.29±0.53	49.82±2.55	61.23
SEPC	30%	**32.80±0.09**	**79.49±0.63**	**60.19±1.91**	**68.74±1.83**	**81.55±0.44**	**72.73±0.19**	**69.79±0.54**	**71.57±0.49**	**44.89±0.71**	**51.49±0.53**	**63.32**

Table 3: Robustness analysis evaluation (%) results against different noise rates. The best results are bolded.

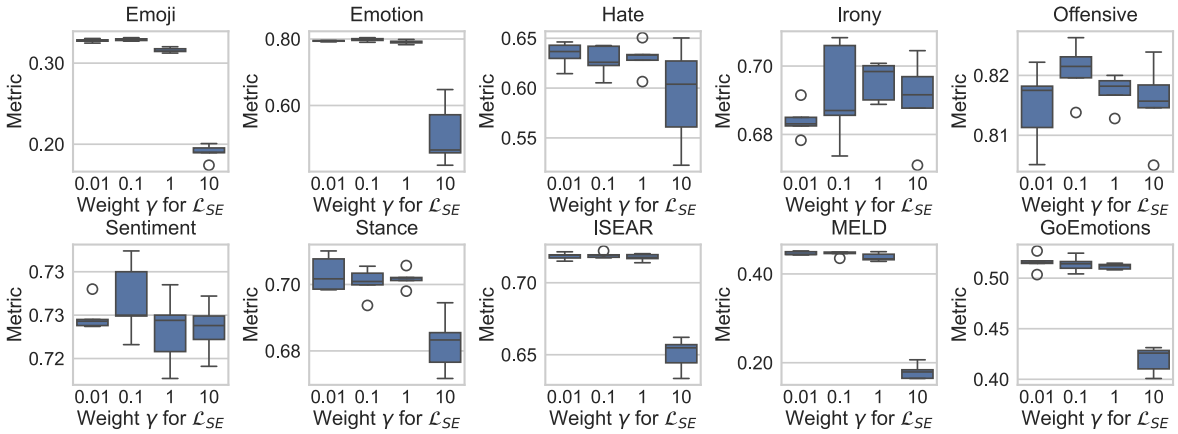


Figure 4: The impact of the weight parameter γ on the regularization loss $\mathcal{L}_{SE}$.

Table 3. SEPC shows superior performance across all noise rate settings and all datasets compared to baselines. Specifically, it outperforms all baselines with average improvements of 2.23%-4.30%, 1.67%-4.64%, and 2.09%-5.14% under 10%, 20%, and 30% noise rates, respectively. Besides, when the noise rate increases from 20% to 30%, SEPC exhibits a minimal average performance decrease. This experiment demonstrates that SEPC has better robustness when handling noise and data unreliability.

Hyperparameter Sensitivity Analysis We evaluate the impact of the newly introduced weight hyperparameter γ for regularization loss $\mathcal{L}_{SE}$ on ten classification datasets and illustrate the results in Figure 4. A lower regularization weight is preferred in most datasets. A too-large weight, $\gamma = 10$, generally leads to a noticeable performance decrement and higher variance.

Generalization Analysis We conduct experiments under limited training data conditions to better evaluate the generalization capability of SEPC. In detail, we randomly select 90%, 70%, 50%, and 30% of the training data during the model training period and compare the performance of SEPC on the test set with other probabilistic coding models. The experimental results are illustrated in Figure 5. SEPC outperforms all other baselines across all datasets under different percentages of the training set. The superior performance under the limited training data demonstrates the generalization capability of SEPC.

Visualization We visualize the embeddings of SEPC, VIB, and SPC on the ISEAR dataset to intuitively showcase the advantages of SEPC's learned embeddings. As shown in Figure 6, the embeddings of SEPC are more discriminative. Additionally, the embedding distribution of SEPC has a larger standard deviation, thus occupying a larger embedding space. This results in better generalization capability.

Related Work

Probabilistic Embedding Compared to deterministic embedding (Dong, Yan, and Wang 2024; Xu et al. 2024), probabilistic embedding learns a probabilistic distribution for each input, effectively capturing data uncertainty and complexity, and thus better handling noise and outliers. The mainstream probabilistic embedding methods follow the Information Bottleneck (IB) principle (Tishby, Pereira, and

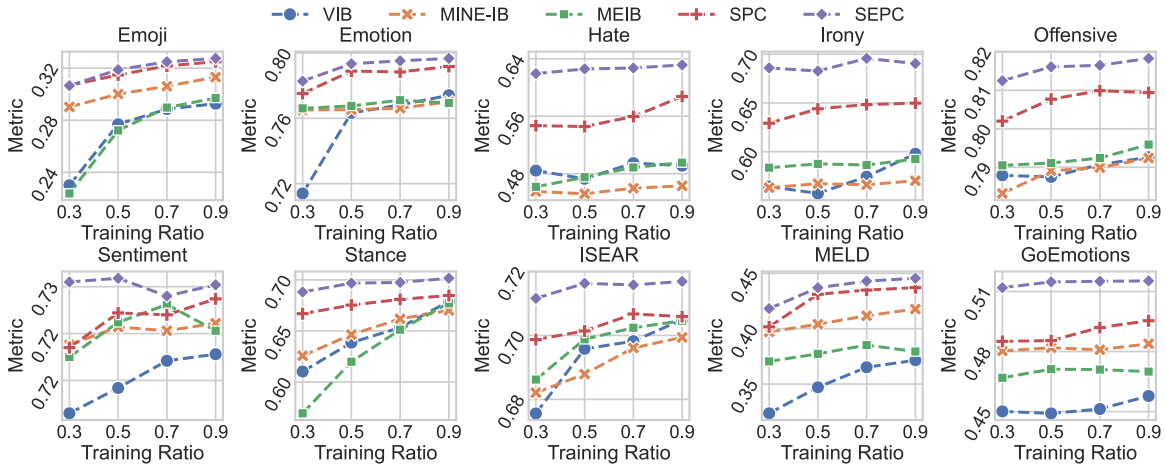


Figure 5: Results of different models with different ratios of the training set.

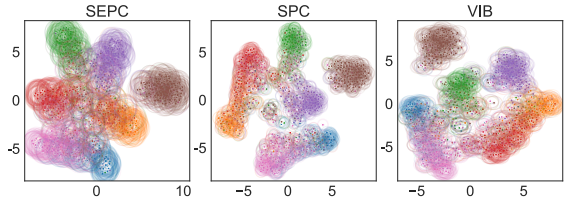


Figure 6: Visualization of embeddings. The circle represents the standard deviation of the probabilistic embeddings.

Bialek 2000; Tishby and Zaslavsky 2015), which seeks to discover compressed representations that retain the maximum amount of relevant information for the prediction task while eliminating as much irrelevant information as possible. VIB (Alemi et al. 2017) constrains the latent variable to follow the Gaussian distribution and utilizes Kullback-Leibler (KL) divergence between the learned distribution and the prior Gaussian distribution as the regularization loss. Sparse IB (Chalk, Marre, and Tkacik 2016) replaces the prior Gaussian distribution of VIB with the Student-t distribution. MINE-IB (Belghazi et al. 2018) is a mutual information neural estimation method with the IB principle, allowing for the tractable IB application in a continuous setting. Fischer (2020) proposes the conditional entropy bottleneck top improved robustness to adversarial examples. MEIB (An, Jammalamadaka, and Chong 2023) utilizes maximum conditional entropy to serve as the bottleneck of IB. SPC (Hu et al. 2024) introduces an encoder-only framework, incorporating a class-level structured regularization loss.

Structural Entropy Unlike early information entropy, such as Shannon entropy, which is defined by unstructured probability distributions, structural entropy (Li and Pan 2016) takes the hierarchical structural information of the input data into account. It is gaining substantial traction and is widely used in graph structural learning (Zou et al. 2023), node classification (Duan et al. 2024), social bot detection (Peng et al. 2024; Zeng, Peng, and Li 2024), and deep clustering (Sun et al. 2024). USER (Wang et al. 2023) proposes a structural entropy-based loss. However, current works focus solely on minimizing structural entropy to maximize task-related information and are limited to classification tasks.

Conclusion

In this paper, we propose SEPC, a structural entropy guided probabilistic coding model. SEPC utilizes maximizing the structural entropy as the regularization loss, introducing the structural information into the optimization, and aims to separate the latent variables in the class space. Additionally, we propose a probabilistic encoding tree and an effective method to utilize the structural entropy for regression tasks based on it. Experiments on 12 datasets demonstrate the effectiveness, generalization capability, and robustness of SEPC in both classification and regression tasks.

Acknowledgments

This work is supported by the National Key R&D Program of China through grant 2022YFB3104703, NSFC through grants 62322202, 62432006, and 62476163, Local Science and Technology Development Fund of Hebei Province Guided by the Central Government of China through grant 246Z0102G, Guangdong Basic and Applied Basic Research Foundation through grant 2023B1515120020, Open Research Fund from Guangdong Laboratory of Artificial Intelligence and Digital Economy (SZ) under Grant No. GML-KF-24-08, and CCF-DiDi GAIA Collaborative Research Funds for Young Scholars.

References

Alemi, A. A.; Fischer, I.; Dillon, J. V.; and Murphy, K. 2017. Deep Variational Information Bottleneck. In *International Conference on Learning Representations (ICLR)*.

An, S.; Jammalamadaka, N.; and Chong, E. 2023. Maximum Entropy Information Bottleneck for Uncertainty-aware

Stochastic Embedding. In *Computer Vision and Pattern Recognition (CVPR)*, 3809–3818.

Barbieri, F.; Camacho-Collados, J.; Ronzano, F.; Anke, L. E.; Ballesteros, M.; Basile, V.; Patti, V.; and Saggion, H. 2018. Semeval 2018 task 2: Multilingual emoji prediction. In *Proceedings of the 12th international workshop on semantic evaluation*, 24–33.

Basile, V.; Bosco, C.; Fersini, E.; Nozza, D.; Patti, V.; Pardo, F. M. R.; Rosso, P.; and Sanguinetti, M. 2019. Semeval-2019 task 5: Multilingual detection of hate speech against immigrants and women in twitter. In *Proceedings of the 13th international workshop on semantic evaluation*, 54–63.

Belghazi, M. I.; Baratin, A.; Rajeswar, S.; Ozair, S.; Bengio, Y.; Hjelm, R. D.; and Courville, A. C. 2018. Mutual Information Neural Estimation. In *International Conference on Machine Learning*, 530–539.

Cer, D.; Diab, M.; Agirre, E.; Lopez-Gazpio, I.; and Specia, L. 2017. SemEval-2017 Task 1: Semantic Textual Similarity Multilingual and Crosslingual Focused Evaluation. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, 1–14.

Chalk, M.; Marre, O.; and Tkacik, G. 2016. Relevant sparse codes with variational information bottleneck. In *Conference on Neural Information Processing Systems (NeurIPS)*, 1957–1965.

Cortes, C.; and Vapnik, V. 1995. Support-vector networks. *Machine learning*, 20: 273–297.

Demszky, D.; Movshovitz-Attias, D.; Ko, J.; Cowen, A.; Nemade, G.; and Ravi, S. 2020. GoEmotions: A Dataset of Fine-Grained Emotions. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 4040–4054.

Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *North American Chapter of the Association for Computational Linguistics (NAACL)*, 4171–4186.

Dong, W.; Yan, D.; and Wang, P. 2024. Self-Supervised Node Representation Learning via Node-to-Neighbourhood Alignment. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46: 4218–4233.

Duan, L.; Chen, X.; Liu, W.; Liu, D.; Yue, K.; and Li, A. 2024. Structural Entropy Based Graph Structure Learning for Node Classification. In *AAAI Conference on Artificial Intelligence (AAAI)*, 8372–8379.

Fischer, I. 2020. The Conditional Entropy Bottleneck. *Entropy*, 22(9): 999.

Gunel, B.; Du, J.; Conneau, A.; and Stoyanov, V. 2021. Supervised Contrastive Learning for Pre-trained Language Model Fine-tuning. In *9th International Conference on Learning Representations*.

Hochreiter, S.; and Schmidhuber, J. 1997. Long short-term memory. *Neural computation*, 9(8): 1735–1780.

Hu, D.; Hou, X.; Du, X.; Zhou, M.; Jiang, L.; Mo, Y.; and Shi, X. 2022. VarMAE: Pre-training of Variational Masked Autoencoder for Domain-adaptive Language Understanding. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 6276–6286.

Hu, D.; Wei, L.; Liu, Y.; Zhou, W.; and Hu, S. 2024. Structured Probabilistic Coding. In *AAAI Conference on Artificial Intelligence (AAAI)*, 12491–12501.

Joulin, A.; Grave, E.; Bojanowski, P.; and Mikolov, T. 2017. Bag of Tricks for Efficient Text Classification. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, 427–431.

Kim, J.; Kim, M.; Woo, D.; and Kim, G. 2021. Drop-Bottleneck: Learning Discrete Compressed Representation for Noise-Robust Exploration. In *International Conference on Learning Representations (ICLR)*.

Kingma, D. P.; and Welling, M. 2013. Auto-Encoding Variational Bayes. In *International Conference on Learning Representations*. Explorandum Ltd.

Li, A.; and Pan, Y. 2016. Structural Information and Dynamical Complexity of Networks. *IEEE Transactions on Information Theory*, 62(6): 3290–3339.

Liu, Y.; Ott, M.; Goyal, N.; Du, J.; Joshi, M.; Chen, D.; Levy, O.; Lewis, M.; Zettlemoyer, L.; and Stoyanov, V. 2020. RoBERTa: A Robustly Optimized BERT Pretraining Approach. *ArXiv*, abs/1907.11692.

Ma, J.; Wang, C.; Liu, Y.; Lin, L.; and Li, G. 2023. Enhanced Soft Label for Semi-Supervised Semantic Segmentation. *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, 1185–1195.

Mahabadi, R. K.; Belinkov, Y.; and Henderson, J. 2021. Variational Information Bottleneck for Effective Low-Resource Fine-Tuning. In *International Conference on Learning Representations (ICLR)*.

Miyato, T.; Dai, A. M.; and Goodfellow, I. J. 2017. Adversarial Training Methods for Semi-Supervised Text Classification. In *5th International Conference on Learning Representations*.

Mohammad, S.; Bravo-Marquez, F.; Salameh, M.; and Kiritchenko, S. 2018. Semeval-2018 task 1: Affect in tweets. In *Proceedings of the 12th international workshop on semantic evaluation*, 1–17.

Mohammad, S.; Kiritchenko, S.; Sobhani, P.; Zhu, X.; and Cherry, C. 2016. Semeval-2016 task 6: Detecting stance in tweets. In *Proceedings of the 10th international workshop on semantic evaluation (SemEval-2016)*, 31–41.

Muthukumar, V.; Narang, A.; Subramanian, V.; Belkin, M.; Hsu, D.; and Sahai, A. 2021. Classification vs regression in overparameterized regimes: Does the loss function matter? *Journal of Machine Learning Research*, 22(222): 1–69.

Oh, S. J.; Murphy, K. P.; Pan, J.; Roth, J.; Schroff, F.; and Gallagher, A. C. 2019. Modeling Uncertainty with Hedged Instance Embeddings. In *International Conference on Learning Representations (ICLR)*.

Peng, H.; Zhang, J.; Huang, X.; Hao, Z.; Li, A.; Yu, Z.; and Yu, P. S. 2024. Unsupervised Social Bot Detection via Structural Information Theory. *TOIS*, 42(6).

Pereyra, G.; Tucker, G.; Chorowski, J.; Kaiser, L.; and Hinton, G. E. 2017. Regularizing Neural Networks by Penalizing Confident Output Distributions. In *5th International Conference on Learning Representations*.

Pintea, S. L.; Lin, Y.; Dijkstra, J.; and van Gemert, J. C. 2023. A step towards understanding why classification helps regression. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 19972–19981.

Poria, S.; Hazarika, D.; Majumder, N.; Naik, G.; Cambria, E.; and Mihalcea, R. 2019. MELD: A Multimodal Multi-Party Dataset for Emotion Recognition in Conversations. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 527–536.

Rosenthal, S.; Farra, N.; and Nakov, P. 2017. SemEval-2017 Task 4: Sentiment Analysis in Twitter. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, 502–518.

Roth, M.; Anthonio, T.; and Sauer, A. 2022. SemEval-2022 Task 7: Identifying Plausible Clarifications of Implicit and Underspecified Phrases in Instructional Texts. In *Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022)*, 1039–1049.

Scherer, K. R.; and Wallbott, H. G. 1994. Evidence for universality and cultural variation of differential emotion response patterning. *Journal of personality and social psychology*, 66(2): 310.

Shi, Y.; and Jain, A. K. 2019. Probabilistic Face Embeddings. In *IEEE International Conference on Computer Vision (ICCV)*, 6901–6910.

Stewart, L.; Bach, F.; Berthet, Q.; and Vert, J.-P. 2023. Regression as classification: Influence of task formulation on neural network features. In *International Conference on Artificial Intelligence and Statistics*, 11563–11582. PMLR.

Sun, L.; Huang, Z.; Peng, H.; Wang, Y.; Liu, C.; and Yu, P. S. 2024. LSEnet: Lorentz Structural Entropy Neural Network for Deep Graph Clustering. In *Forty-first International Conference on Machine Learning*.

Sun, Q.; Li, J.; Peng, H.; Wu, J.; Fu, X.; Ji, C.; and Yu, P. S. 2022. Graph Structure Learning with Variational Information Bottleneck. In *AAAI Conference on Artificial Intelligence (AAAI)*, 4165–4174.

Tishby, N.; Pereira, F. C.; and Bialek, W. 2000. The information bottleneck method. *ArXiv*, physics/0004057.

Tishby, N.; and Zaslavsky, N. 2015. Deep learning and the information bottleneck principle. In *Information Theory Workshop*, 1–5.

Van Hee, C.; Lefever, E.; and Hoste, V. 2018. Semeval-2018 task 3: Irony detection in english tweets. In *Proceedings of the 12th international workshop on semantic evaluation*, 39–50.

Vilnis, L.; and McCallum, A. 2015. Word Representations via Gaussian Embedding. In *International Conference on Learning Representations (ICLR)*.

Wang, Y.; Wang, Y.; Zhang, Z.; Yang, S.; Zhao, K.; and Liu, J. 2023. USER: Unsupervised Structural Entropy-Based Robust Graph Neural Network. In *AAAI Conference on Artificial Intelligence (AAAI)*, 10235–10243. Association for the Advancement of Artificial Intelligence (AAAI).

Wu, J.; Chen, X.; Shi, B.; Li, S.; and Xu, K. 2023. SEGA: Structural Entropy Guided Anchor View for Graph Contrastive Learning. In *International Conference on Machine Learning (ICML)*, volume abs/2305.04501, 37293–37312.

Xu, Z.; Wu, D.; Yu, C.; Chu, X.; Sang, N.; and Gao, C. 2024. SCTNet: Single-Branch CNN with Transformer Semantic Information for Real-Time Segmentation. In *AAAI Conference on Artificial Intelligence (AAAI)*, 6378–6386.

Zampieri, M.; Malmasi, S.; Nakov, P.; Rosenthal, S.; Farra, N.; and Kumar, R. 2019. SemEval-2019 Task 6: Identifying and Categorizing Offensive Language in Social Media (OffensEval). In *Proceedings of the 13th International Workshop on Semantic Evaluation*, 75–86.

Zeng, X.; Peng, H.; and Li, A. 2023. Effective and Stable Role-Based Multi-Agent Collaboration by Structural Information Principles. In *AAAI Conference on Artificial Intelligence (AAAI)*, 11772–11780.

Zeng, X.; Peng, H.; and Li, A. 2024. Adversarial socialbots modeling based on structural information principles. In *Proceedings of the AAAI*, volume 38, 392–400.

Zou, D.; Peng, H.; Huang, X.; Yang, R.; Li, J.; Wu, J.; Liu, C.; and Yu, P. S. 2023. SE-GSL: A General and Effective Graph Structure Learning Framework through Structural Entropy Optimization. In *Proceedings of the ACM Web Conference 2023*.