# An Information Bottleneck Approach for Controlling Conciseness in Rationale Extraction

**Bhargavi Paranjape**[†] **Mandar Joshi**[†] **John Thickstun**[†]
**Hannaneh Hajishirzi**[†ϵ] **Luke Zettlemoyer**[†]
[†] Allen School of Computer Science & Engineering, University of Washington, Seattle, WA
[ϵ]Allen Institute of Artificial Intelligence, Seattle
{bparan,mandar90,thickstn,hannaneh,lsz}@cs.washington.edu

## Abstract

Decisions of complex models for language understanding can be explained by limiting the inputs they are provided to a relevant subsequence of the original text — a *rationale*. Models that condition predictions on a concise rationale, while being more interpretable, tend to be less accurate than models that are able to use the entire context. In this paper, we show that it is possible to better manage the trade-off between concise explanations and high task accuracy by optimizing a bound on the Information Bottleneck (IB) objective. Our approach jointly learns an explainer that predicts sparse binary masks over input sentences without explicit supervision, and an end-task predictor that considers only the residual sentences. Using IB, we derive a learning objective that allows direct control of mask sparsity levels through a tunable sparse prior. Experiments on the ERASER benchmark demonstrate significant gains over previous work for both task performance and agreement with human rationales. Furthermore, we find that in the semi-supervised setting, a modest amount of gold rationales (25% of training examples with gold masks) can close the performance gap with a model that uses the full input.[1]

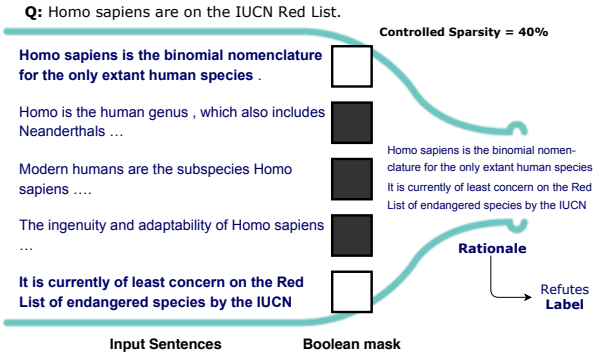


Figure 1: Our Information Bottleneck-based approach extracts concise rationales that are minimally informative about the original input, and maximally informative about the label through fine-grained control of sparsity in the bottleneck (0.4 in this fact verification example). End-task prediction is conditioned only on the bottlenecked input.

## 1 Introduction

A rationale is a short yet sufficient part of the input text that can explain model decisions for a range of language understanding tasks (Lei et al., 2016). Models can be *faithful* to a rationale by only using the selected text as input for end-task prediction (DeYoung et al., 2019). However, there is almost always a trade-off between interpretable models that learn to extract *sparse* rationales and more *accurate* models that are able to use the full context but provide little explanation for their predictions (Lei et al., 2016; Weld and Bansal, 2019). In this paper, we show that it is possible to better manage this trade-off by optimizing a novel bound on the Information Bottleneck (Tishby et al., 1999) objective (Figure 1).

We follow recent work in representing rationales as binary masks over the input text (Lei et al., 2016; Bastings et al., 2019). During learning, it is common to encourage sparsity by minimizing a norm on the rationale masks (e.g. $L_0$ or $L_1$) (Lei et al., 2016; Bastings et al., 2019). It is often challenging to control the sparsity-accuracy trade-off in norm-minimization methods; we show that these methods seem to push too directly for sparsity at the expense of accuracy (Section 5.2). Our approach, in contrast, allows more control through a prior that specifies task-specific target sparsity levels that should be met in expectation across the training set.

More specifically, we formalize the problem of

[1]Our code is available at https://github.com/bhargaviparanjape/explainable_qa

inducing controlled sparsity in the mask using the Information Bottleneck (IB) principle. Our approach seeks to extract a rationale as an optimal *compressed* intermediate representation (the bottleneck) that is both (1) minimally informative about the original input, and (2) maximally informative about the output class. We derive a novel variational bound on the IB objective for our case where we constrain the intermediate representation to be a concise subsequence of the input, thus ensuring its interpretablity.

Our model consists of an *explainer* that extracts a rationale from the input, and an *end-task predictor* that predicts the output based only on the extracted rationale. Our IB-based training objective guarantees sparsity by minimizing the Kullback–Leibler (KL) divergence between the explainer mask probability distribution and a prior distribution with controllable sparsity levels. This prior probability affords us tunable fine-grained control over sparsity, and allows us to bias the proportion of the input to be used as rationale. We show that, unlike norm-minimization methods, our KL-divergence objective is able to consistently extract rationales with the specified sparsity levels.

Across five tasks from the ERASER interpretability benchmark (DeYoung et al., 2019) and the BeerAdvocate dataset (McAuley et al., 2012), our IB-based sparse prior objective has significant gains over previous norm-minimization techniques — up to 5% relative improvement in task performance metrics and 6% to 80% relative improvement in agreement with human rationale annotations. Our interpretable model achieves task performance within 10% of a model of comparable size that uses the entire input. Furthermore, we find that in the semi-supervised setting, adding a small proportion of gold rationale annotations (approximately 25% of the training examples) bridges this gap — we are able to build an interpretable model without compromising performance.

## 2 Method

### 2.1 Task and Method Overview

We assume supervised text classification or regression data that contains tuples of the form $(x, y)$. The input document $x$ can be decomposed into a sequence of sentences $x = (x_1, x_2, \ldots, x_n)$ and $y$ is the category, answer choice, or target value to predict. Our goal is to learn a model that not only predicts $y$, but also extracts a rationale or explanation $z$—a latent *subsequence* of sentences in $x$ with the following properties:

1. Model prediction $y$ should rely entirely on $z$ and not on its complement $x \backslash z$ — *faithfulness* (DeYoung et al., 2019).
2. $z$ must be concise, i.e., it should contain as few sentences as possible without sacrificing the ability to correctly predict $y$.

Following Lei et al. (2016), our interpretable model learns a boolean mask $m = (m_1, m_2, \ldots, m_n)$ over the sentences in $x$, where $m_j \in \{0, 1\}$ is a discrete binary variable. To enforce (1), the masked input $z = m \odot x = (m_1 \cdot x_1, m_2 \cdot x_2, \ldots, m_n \cdot x_n)$ is used to predict $y$. Conciseness is attained using an information bottleneck.

### 2.2 Formalizing Interpretability Using Information Bottleneck

**Background** The Information Bottleneck (IB) method is used to learn an optimal compression model that transmits information from a random variable $X$ to another random variable $Y$ through a compressed representation $Z$. The IB objective is to minimize the following:

$$L_{IB} = I(X, Z) - \beta I(Z, Y), \tag{1}$$

where $I(\cdot, \cdot)$ is mutual information. This objective encourages $Z$ to only retain as much information about $X$ as is needed to predict $Y$. The hyperparameter $\beta$ controls the trade-off between retaining information about either $X$ or $Y$ in $Z$. Alemi et al. (2016) derive the following variational bound on Equation 1:[2]

$$L_{VIB} = \underbrace{\mathbb{E}_{z \sim p_\theta(z|x)}[-\log q_\phi(y|z)]}_{\text{Task Loss}} + \underbrace{\beta KL[p_\theta(z|x), r(z)]}_{\text{Information Loss}}, \tag{2}$$

where $q_\phi(y|z)$ is a parametric approximation to the true likelihood $p(y|z)$; $r(z)$, the prior probability of $z$, approximates the marginal $p(z)$; and $p_\theta(z|x)$ is the parametric posterior distribution over $z$.

The information loss term in Equation 2 reduces $I(X, Z)$ by decreasing the KL divergence[3] be-

[2]For brevity and clarity, objectives are shown for a single data point. More details of this bound can be found in Appendix A.1 and Alemi et al. (2016).

[3]To analytically compute the KL-divergence term, the posterior and prior distributions over $z$ are typically K-dimensional multivariate normal distributions. Compression is achieved by setting $K << D$, the input dimension of $X$.

tween the posterior distribution $p_\theta(z|x)$ that depends on $x$ and a prior distribution $r(z)$ that is independent of $x$. The task loss encourages predicting the correct label $y$ from $z$ to increase $I(Z, Y)$.

**Our Variational Bound for Interpretability** The learned bottleneck representation $z$, found via Equation 2, is not human-interpretable as $z$ is typically a compressed continuous vector representation of input $x$.[3] To ensure interpretability of $z$, we define the interpretable latent representation as $z := m \odot x$, where $m$ is a boolean mask on the input sentences in $x$. We assume that the mask variables $m_j$ over individual sentences are conditionally independent given the input $x$, i.e. the posterior $p_\theta(m|x) = \prod_j p_\theta(m_j|x)$, where $p_\theta(m_j|x) = \text{Bernoulli}(\theta_j(x))$ and $j$ indexes sentences in the input text.[4] Because $z := m \odot x$, the posterior distribution over $z$ is a mixture of dirac-delta distributions:

$$p_\theta(z_j|x) = (1 - \theta_j(x))\delta(z_j) + \theta_j(x)\delta(z_j - x_j),$$

where $\delta(x - c)$ is the dirac-delta probability distribution that is zero everywhere except at $c$.

For the prior, we assume a fixed Bernoulli distribution over mask variables. For instance, $r(m_j) = \text{Bernoulli}(\pi)$ for some constant $\pi \in (0, 1)$. This also induces a fixed distribution on $z$ via the definition $z := m \odot x$. Instead of using an expressive $r(z)$ to approximate $p(z)$, we use a non-parametric prior $r(z)$ to force the marginal $p(z)$ of the learned distribution over $z$ to approximately equal $\pi$. Our characterization of the prior and the posterior achieves compression of the input via *sparsity* in the latent representation, in contrast to compression via dimensionality reduction (Alemi et al., 2016).

For the intermediate representation $z := m \odot x$, we can decompose $\text{KL}(p_\theta(z_j|x), r(z_j))$ as:

$$\text{KL}(p_\theta(m_j|x), r(m_j)) + \pi H(x)$$

Since the entropy of the input, $\pi H(x)$, is a constant with respect to $\theta$, it can be dropped. Hence, we obtain the following variational bound on IB with interpretability constraints over $z$, derived in more detail in Appendix A.2:

$$L_{IVIB} = \mathbf{E}_{m \sim p_\theta(m|x)}[-\log q_\phi(y|m \odot x)] + \beta \sum_j KL[p_\theta(m_j|x)||r(m_j)] \quad (3)$$

[4] We use Bernoulli distribution for $m_j$ in this work, but any binary distribution for which KL divergence can be analytically computed can be used.

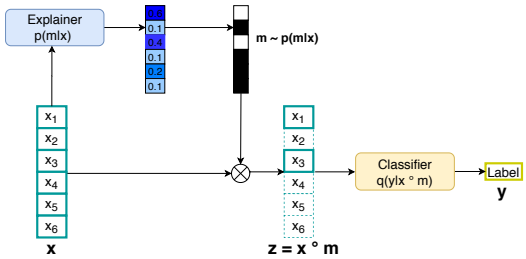


Figure 2: Architecture: The explainer extracts a rationale from the input using a binary mask, and an end-task predictor predicts the output based only on the extracted rationale.

The first term is the expected cross-entropy term for the task which can be computed by drawing samples $m \sim p_\theta(m|x)$. The second information-loss term encourages the mask $m$ to be independent of $x$ by reducing the KL divergence of its posterior $p_\theta(m|x)$ from a prior $r(m)$ that is independent of $x$. However, this does not necessarily remove information about $x$ in $z = x \odot m$. For instance, a mask consisting of all ones is independent of $x$, but in this case $z = x$ and the rationale is no longer concise. In the following section, we present a simple way to avoid this degenerate case in practice by appropriately fixing the value of $\pi$.

### 2.3 The Sparse Prior Objective

The key to ensuring that $z = m \odot x$ is strictly a subsequence of $x$ lies in the fact that $r(m_j) = \pi$ is our prior belief about the probability of a sentence being important for prediction. For instance, if humans annotate $10\%$ of the input text as a rationale, we can fix our prior belief that a sentence should be a part of the mask as $r(m_j) = \pi = 0.1\ \forall j$. IB allows us to *control* the amount of sparsity in the mask that is eventually sampled from the learned distribution $p_\theta(m|x)$ in several ways. $\pi$ can be estimated as the expected sparsity of the mask from expert rationale annotations. If such a statistic is not available, it can be explicitly tuned for the desired trade-off between end task performance and rationale length. In this work, we assume $\pi \in (0, 0.5)$ so that the sampled mask is sparse. We refer to this training objective with tunable $r(m) = \pi$ as the sparse prior (Sparse IB) method in our experiments. In Appendix C, we also discuss explicitly learning the value of $\pi$.

## 3 Model

To optimize for objective 3, the posterior distribution estimator $p_\theta()$ and label likelihood estimator

$q_\phi()$ are instantiated as the *explainer* and *end-task predictor* neural models respectively. Two different pre-trained transformers (Devlin et al., 2019) are used to initialize both models.

### 3.1 Architecture

**Explainer** $p_\theta(z|x)$**:** Given an input $x = x_1, x_2, \ldots, x_n$ consisting of $n$ sentences, the explainer produces a binary mask $m \in \{0,1\}^n$ over the input sentences which is used to derive a rationale $z = m \odot x$. It maps every sentence $x_j$ to its probability, $p_\theta(m_j|x)$ of being selected as part of $z$ where $p(\cdot)$ is a binary distribution. The explainer contextualizes the input sequence $x$ at the token level, and produces sentence representations $\mathbf{x} = (\mathbf{x_1}, \mathbf{x_2}, \ldots, \mathbf{x_n})$ where $\mathbf{x_j}$ is obtained by concatenating the contextualized representations of the first and last tokens in sentence $x_j$. A linear layer is used to transform these representations into logits (log probabilities) of a Bernoulli distribution. We choose the Bernoulli distribution since its sample can be reparameterized as described in Section 3.2, and we can analytically compute the KL-divergence term between two Bernoulli distributions. In Appendix C, we also experiment with the Kumaraswamy distribution (Fletcher and Ponnambalam, 1996) used in (Bastings et al., 2019). The mask $m \in \{0,1\}^n$ is constructed by independently sampling each $m_j$ from $p(m_j|x)$.

**End-task Predictor** $q_\phi(y|z)$**:** We define $z$ as the rationale representation $z = m \odot \mathbf{x}$, an element-wise dot product between $m_j$ and the corresponding sentence representation $\mathbf{x_j}$. The end-task predictor uses $z$ to predict the output variable $y$. The same hard attention mask $m$ is applied to all end-task transformer layers at every head to ensure prediction relies only on $m \odot x$. The predictor further consists of a log-linear classifier layer over the `[CLS]` token, similar to Devlin et al. (2019). When an optional query sequence is available for datasets like BoolQ, we do not mask it as it is assumed to be essential to predict $y$ (see Appendix B.2 for implementation details).

### 3.2 Training and Inference

The sampling operation of the discrete binary variable $m_j \in \{0,1\}$ in Section 3.1 is not differentiable. Lei et al. (2016) use a simple Bernoulli distribution with REINFORCE (Williams, 1992) to overcome non-differentiability. We found REINFORCE to be quite unstable with high variance in results. Instead, we employ reparameterization (Kingma et al., 2015) to facilitate end-to-end differentiability of our approach. We use the Gumbel-Softmax reparameterization (Jang et al., 2017) for categorical (here, binary) distributions to reparameterize the Bernoulli variables $m_j$. The reparameterized binary variable $m_j^*$ is generated as follows:

$$m_j^* = \sigma\left(\frac{\log p(m_j|x) + g_j}{\tau}\right),$$

where $\sigma$ is the Sigmoid function, $\tau$ is a hyperparameter for the temperature of the Gumbel-Softmax function, and $g_j$ is a random sample from the Gumbel(0,1) distribution (Gumbel, 1948). $m_j^* \in (0,1)$ is a continuous and differentiable approximation to $m_j$ with low variance.

During inference, we extract the top $\pi\%$ sentences with largest $p_\theta(m_j|x)$ values, where $\pi$ corresponds to the threshold hyperparameter described in Section 2.3. Previous work (Lei et al., 2016; Bastings et al., 2019) samples from $p(m|x)$ during inference. Such an inference strategy is non-deterministic, making comparison of different masking strategies difficult. Moreover, it is possible to appropriately scale $p(m_j|x)$ values to obtain better inference results, thereby not reflecting if $p(m_j|x)\ \forall j$ are correctly ordered. By allowing a fixed budget of $\pi\%$ per example, we are able to fairly compare approaches in Section 4.3.

### 3.3 Semi-Supervised Setting

As we will show in Section 5, despite better control over the sparsity-accuracy trade-off, there is still a gap in task performance between our unsupervised approach and a model that uses full context. To bridge this gap and better manage the trade-off at minimal annotation cost, we experiment with a semi-supervised setting where we have annotated rationales for part of the training data.

For input example $x = (x_1, x_2, \ldots, x_n)$ and a gold mask $\hat{m} = (\hat{m_1}, \hat{m_2}, \ldots, \hat{m_n})$ over sentences, we use the following semi-supervised objective:

$$L_{semi} = \mathbf{E}_{m\sim p_\theta(m|x)}[-\log q(y|m \odot x)] + \gamma \sum_j -\hat{m_j} \log p(m_j|x) \quad (4)$$

While we still sample from $p(m|x)$ and train end-to-end using reparameterization, the information loss over $p(m|x)$ is replaced with the supervised rationale loss.

## 4 Experimental Setup

### 4.1 End Tasks

We evaluate our Sparse IB approach on five text classification tasks from the ERASER benchmark (DeYoung et al., 2019) and the BeerAdvocate regression task (McAuley et al., 2012) used in Lei et al. (2016).

**ERASER**: The ERASER tasks we evaluate on include the Movies sentiment analysis task (Pang and Lee, 2004), the FEVER fact extraction and verification task (Thorne et al., 2018), the MultiRC (Khashabi et al., 2018) and BoolQ (Clark et al., 2019) reading comprehension tasks, and the Evidence Inference classification task (Lehman et al., 2019) over scientific articles for results of medical interventions.

**BeerAdvocate** (McAuley et al., 2012): The BeerAdvocate regression task for predicting 0-5 star ratings for multiple aspects like appearance, smell, and taste based on reviews.

All these datasets have *sentence-level* rationale annotations for validation and test sets. We do not consider e-SNLI (Camburu et al., 2018) and CoS-E (Rajani et al., 2019) in ERASER as they have only 1-2 input sentences, rationales annotations at word level, and often require common sense/world knowledge. The ERASER tasks contain rationale annotations for the training set, which we only use for our semi-supervised experiments. We closely follow dataset processing in the ERASER benchmark setup and Bastings et al. (2019) (for BeerAdvocate). Additionally, for BoolQ and Evidence Inference which contain longer documents, we use a sliding window to select a single document *span* that has the maximum TF-IDF score against the question (further details in Appendix B.1).

### 4.2 Setup

**Evaluation Metrics** We adopt the metrics proposed for the ERASER benchmark to evaluate both agreement with comprehensive human rationales as well as end task performance. To evaluate quality of rationales, we report the token-level Intersection-Over-Union F1 (IOU F1), which is a relaxed measure for comparing two sets of text spans. We also report token-level F1 scores. For task accuracy, we report weighted F1 for classification tasks, and the mean square error for the BeerAdvocate regression task.

**Implementation Details** We use BERT-base with a maximum context-length of 512 to instantiate the combined explainer and end-task predictor. Models are tuned on the development set using the rationale IOU F1. Appendix B.3 contains details about hyperparameters.

### 4.3 Baselines

We first consider two bounding scenarios where no rationales are predicted. In the **Full Context (Full)** setting, the entire context is used to make predictions; this allows us to estimate the loss in performance as a result of interpretable hard attention models that only use $\pi\%$ of the input. In the **Gold Rationale (Gold)** setting, we train a model to only use human rationale annotations during *training and inference* to estimate an upper-bound on task and rationale performance metrics. We compare our Sparse IB approach with the following baselines. For fair comparison, all baselines are modified to use BERT-based representations.

**Norm Minimization (Sparse Norm)** Existing approaches (Lei et al., 2016; Bastings et al., 2019) learn sparse masks over the inputs by minimizing the $L_0$ norm of the mask $m$ as follows:

$$L_{SL0} = \mathbf{E}_{m\sim p(m|x)}[-\log q(y|z)] + \lambda||m|| \quad (5)$$

Here, $\lambda$ is the weight on the norm.

**Controlled Norm Minimization (Sparse Norm-C)** For fair comparison against our approach for controlled sparsity, we modify Equation 5 to ensure that the norm of $m$ is not penalized when it drops below the threshold $\pi$.

$$L_{SL0-C} = \mathbf{E}_{m\sim p(m|x)}[-\log q(y|z)] + \lambda \max(0, ||m|| - \pi) \quad (6)$$

This modification has also been adopted in recent. work (Jain et al., 2020). Explicit control over sparsity in the mask $m$ through the tunable prior probability $\pi$ naturally emerges from IB theory, as opposed to the modification adopted in norm-based regularization (Equation 6).

**No Sparsity** This method only optimizes for the end-task performance without any sparsity-inducing loss term, to evaluate the effect of sparsity inducing objectives in Sparse IB, Sparse Norm, and Sparse Norm-C.

| Approach | FEVER | | | MultiRC | | | Movies | | |
|---|---|---|---|---|---|---|---|---|---|
| | Task | Token F1 | IOU | Task | Token F1 | IOU | Task | Token F1 | IOU |
| 1. Full | 89.5 | 33.7 | 36.2 | 66.8 | 29.1 | 29.2 | 91.0 | 35.1 | 47.3 |
| 2. Gold | 91.8 | - | - | 76.6 | - | - | 97.0 | - | - |
| Unsupervised | | | | | | | | | |
| 3. No Sparsity | 82.8 | 35.7 | 38.1 | 60.1 | 20.8 | 19.8 | 78.2 | 24.6 | 37.9 |
| 4. Sparse Norm | 83.1 | 40.9 | 44.0 | 59.7 | 19.9 | 20.4 | 78.6 | 23.5 | 34.7 |
| 5. Sparse Norm-C | 83.3 | 41.6 | 44.9 | 61.7 | 21.7 | 21.8 | 81.8 | 22.8 | 34.4 |
| 6. Sparse IB (Us) | **84.7** | **42.7** | **45.5** | **62.1** | **24.9** | **24.3** | **84.0** | **27.5** | **39.6** |
| Supervised | | | | | | | | | |
| 7. Bert-To-Bert (Reported) | 87.7 | 81.2 | 83.5 | 62.4 | 39.9 | 40.9 | 82.4 | 14.5 | 7.5 |
| 8. Bert-To-Bert (Ours$^{\epsilon}$) | 85.0 | 78.1 | **81.7** | 63.3 | 41.2 | 41.6 | 86.0 | 16.2 | 15.7 |
| 9. 25% data (Us) | **88.8** | 63.9 | 66.6 | **66.4** | **54.0** | **54.4** | **85.4** | **28.2** | **43.4** |

| | BoolQ | | | Evidence Inference | | | BeerAdvocate | | |
|---|---|---|---|---|---|---|---|---|---|
| | Task | Token F1 | IOU | Task | Token F1 | IOU | Task | Token F1 | IOU |
| 1. Full | 65.6 | 11.8 | 15.0 | 52.1 | 6.4 | 9.7 | .015 | 38.4 | 37.8 |
| 2. Gold | 85.9 | - | - | 71.7 | - | - | - | - | - |
| Unsupervised | | | | | | | | | |
| 3. No Sparsity | 62.5 | 8.1 | 10.7 | 43.0 | 6.1 | 09.0 | .018 | 48.2 | 47.3 |
| 4. Sparse Norm | 62.5 | 8.5 | 12.8 | 38.9 | 3.4 | 6.3 | .017 | 28.6 | 35.5 |
| 5. Sparse Norm-C | 63.7 | 10.7 | 14.3 | 44.7 | 5.1 | 8.0 | .018 | 49.3 | 49.0 |
| 6. Sparse IB (Us) | **65.2** | **12.8** | **16.5** | **46.3** | **6.9** | **10.0** | **.016** | **53.1** | **52.3** |
| Supervised | | | | | | | | | |
| 8. Bert-To-Bert (Reported) | 54.4 | 13.4 | 5.2 | 70.8 | 46.8 | 45.5 | | | |
| 7. Bert-To-Bert (Ours$^{\epsilon}$) | 62.3 | 12.4 | 31.5 | **70.8** | **54.8** | **53.9** | | $^{\dagger}$ | |
| 9. 25% data (Us) | **63.4** | **15.7** | **32.3** | 46.7 | 10.8 | 13.3 | | | |

Table 1: Task, Rationale IOU F1 (threshold set to 0.1) and Token F1 for our hard-attention Sparse IB approach and baselines on test sets, averaged over 5 random seeds. We report MSE for BeerAdvocate, hence lower is better. Gold IOU and token F1 are 100.0. We use 25% training data in our semi-supervised setting (Section 3.3). Validation set results can be found in Table 6 in the Appendix. $^{\epsilon}$ We could not reproduce numbers for the Bert-to-Bert supervised method reported in DeYoung et al. (2019). $^{\dagger}$ No rationale supervision available for BeerAdvocate.

**Supervised Approach (Pipeline)** Lehman et al. (2019) learn an explainer and a task predictor independently in sequence using supervision for rationales and task labels, using the output of the explainer in the predictor during inference. We compare our semi-supervised model (Section 3.3) with this *pipeline* approach.

## 5 Results

### 5.1 Quantitative Evaluation

Table 1 compares our Sparse IB approach against baselines (Section 4.3). Sparse IB outperforms norm-minimization approaches (rows 4-6) in both agreement with human rationales and task performance across all tasks. We perform particularly well on rationale extraction with 5 to 80% relative improvements over the better performing norm-minimization variant Sparse Norm-C. Sparse IB also attains task performance within 0.5 to 10% of the full-context model (row 1), despite using $< 50\%$ of the input sentences. All unsupervised approaches still obtain a lower IOU F1 compared to the full context model for Movies and MultiRC, primarily due to their considerably lower precision on these benchmarks.

Our results also highlight the importance of explicit *controlled sparsity* inducing terms as effective inductive biases for improved task performance and rationale agreement. Specifically, sparsity-inducing methods consistently outperform the No Sparsity-baseline (row 3). One way to interpret this result is that sparsity objectives add input-dimension regularization during training, which results in better generalization during inference. Moreover, Sparse Norm-C, which adds the element of *control* to norm-minimization, performs considerably better than Sparse Norm. Finally, we see a positive correlation between task performance and

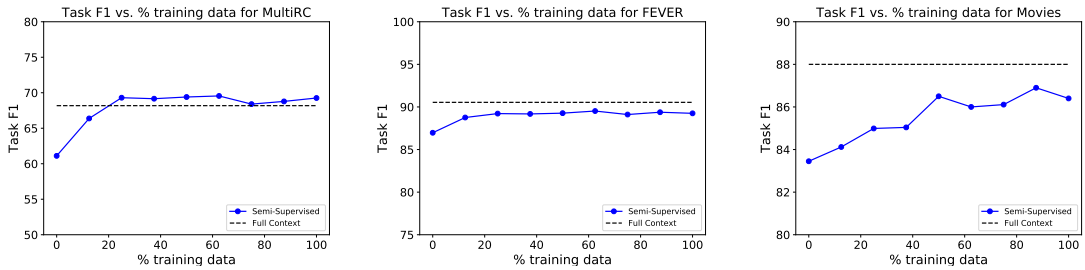


Figure 3: Semi-supervised experiments showing the task performance for varying proportions of rationale annotation supervision on the MultiRC, FEVER, and Movies datasets.

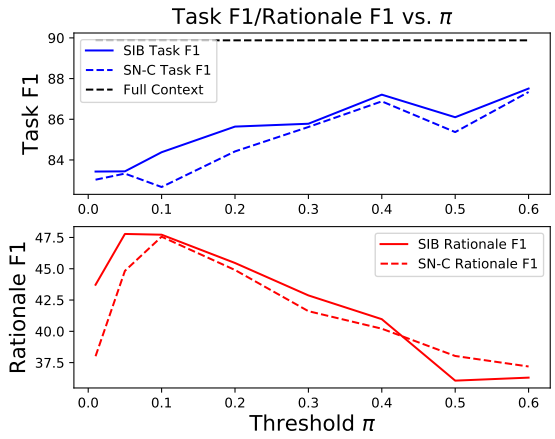


Figure 4: Effect of varying the sparsity hyperparameter $\pi$ to control the trade-off between compactness of rationales and accuracy for the FEVER dataset (right). SIB is Sparse IB and SN-C is Sparse Norm-C.

| Dataset | $\pi$ | Sparse Norm-C Mean | Sparse Norm-C Var | Sparse IB Mean | Sparse IB Var |
|---|---|---|---|---|---|
| FEVER | 0.20 | 0.17 | 0.94 | 0.21 | 1.24 |
| MultiRC | 0.25 | 0.11 | 1.14 | 0.26 | 1.67 |
| Movies | 0.40 | 0.38 | 2.90 | 0.42 | 3.02 |
| BoolQ | 0.20 | 0.04 | 0.84 | 0.22 | 1.91 |
| Evidence | 0.20 | 0.10 | 1.17 | 0.20 | 1.61 |

Table 2: Average mask length (sparsity) attained by Sparse IB and the Sparse Norm-C baseline for a given prior $\pi$ for different tasks, averaged over 100 runs. Mean is reported as the average proportion of sentences to compare with expected sparsity ($\pi$) and variance is reported in the number of sentences.

agreement with human rationales. This is important since accurate models that also better emulate human rationalization likely engender more trust.

**Semi-supervised Setting** In order to close the performance gap with the full-context model, we also experiment with a setup where we minimize the task and the rationale prediction loss using rationale annotations available for a part of the training data (Section 3.3). Figure 4 (left, center) shows the effect of incorporating an increasing proportion of rationale annotation supervision for the FEVER and MultiRC datasets. Our semi-supervised model is even able to match the performance of the full-context models for both FEVER and MultiRC with only 25% of rationale annotation supervision. Furthermore, Figure 4 also shows that these gains can be achieved with relatively modest annotation costs since adding more rationale supervision to the training data seems to have diminishing returns.

Table 1 compares our interpretable model (row 9), which uses rationale supervision for 25% of the training data, with the full-context model and the Pipeline approach (row 8). On three (FEVER, MultiRC, and BoolQ) out of five datasets for which rationale supervision is available, our interpretable models match the task performance of the full-context models while recording large gains in IOU (17-30 F1 absolute). Our approach outperforms the pipeline-based approach in task performance (for FEVER, MultiRC, Movies, and BoolQ) and IOU (for MultiRC and Movies). These gains may result from better exploration due to sampling and inference based on a fixed budget of $\pi\%$ sentences. Our weakest results are on Evidence Inference where the TF-IDF preprocessing often fails to select relevant rationale spans and the pipeline approach uses SciBERT (Beltagy et al., 2019).[5] Our overall results suggest that a small proportion of direct supervision can help build interpretable models without compromising task performance.

## 5.2 Analysis

**Accurate Sparsity Control** Table 2 compares average sparsity rates in rationales extracted by Sparse IB with those extracted by norm-minimization methods. We measure the spar-

[5] Only 51.8% of the selected passages have gold rationales.

| Examples from Error Analysis |
|---|
| Prediction:Positive<br>Ground Truth:Negative<br>The original Babe gets my vote as the best family film since the princess bride, and it's sequel has been getting rave reviews from most internet critics, both Siskel and Ebert sighting it more than a month ago as one of the year's finest films. So, naturally, when I entered the screening room that was to be showing the movie and there was nary another viewer to be found, this notion left me puzzled. It is a rare thing for a children's movie to be praised this highly . . . Looking back, I should have taken the hint and left right when I entered the theater. Believe me; I wanted to like Babe: Pig in the City. The plot seemed interesting enough; . . . It is here that we meet an array of eccentric characters, the most memorable being the family of chimps led by Steven Wright. Here is where the film took a wrong turn . . . unfortunately, the story wears thin as we are introduced to a new set of animals that . . . the main topic of discussion . . . it just didn't feel right and was more painful to watch than it was funny or entertaining, and the same goes for the rest of the movie. |
| Statement : Unforced labor is a reason for human trafficking.<br>Prediction: SUPPORTS<br>Ground Truth: REFUTES<br>DOC: Human trafficking is the trade of humans, most commonly for the purpose of *forced* labour, sexual slavery, or comm--ercial sexual exploitation for the trafficker or others. This may encompass providing a spouse in the context of forced marriage, or the extraction of organs or tissues, including for surrogacy and ova removal. Human trafficking can occur within a country or transnationally. coercion and because of their commercial exploitation . . . In 2012, the I.L.O. estimated that 21 million victims are trapped in modern-day slavery . . . |
| Statement: Atlanta metropolitan area is located in south Georgia.<br>Prediction: SUPPORTS<br>Ground Truth:REFUTES<br>DOC: Metro Atlanta , designated by the United States Office of Management and Budget as the Atlanta-Sandy Springs-Roswell, GA Metropolitan Statistical Area, is the most populous metro area in the US state of Georgia and the ninth-largest metropolitan statistical area (MSA) in the United States. Its economic, cultural and demographic center is Atlanta, and it had a 2015 estimated population of 5.7 million people according to the U.S. Census Bureau. The metro area forms the core of a broader trading area, the Atlanta – Athens-Clarke – Sandy Springs Combined Statistical Area. The Combined Statistical Area spans up to 39 counties in north Georgia and had an estimated 2015 population of 6.3 million people. Atlanta is considered an " alpha world city ". It is the third largest metropolitan region in the Census Bureau's Southeast region behind Greater Washington and South Florida. |

Table 3: Misclassified examples from the Movies and FEVER datasets show: (a) limitations in considering more complex linguistic phenomena like sarcasm; (b) overreliance on shallow lexical matching—*un*forced vs. forced; (c) limited world knowledge—south Georgia, Southeast region, South Florida. *Legend*: Model evidence, Gold evidence, Model and Gold Evidence

sity achieved by the explainer during inference by computing the average number of *one* entries in the input mask $m$ over sentences (the hamming weight) for 100 runs. Sparse IB consistently achieves the sparsity level $\pi$ used in the prior while the norm-minimization approach (Sparse Norm-C) converges to a lower average sparsity for the mask.

**Sparsity-Accuracy Trade-off** Figure 4 (right) shows the variation in task and rationale agreement performance as a function of the sparsity rate $\pi$ for Sparse IB and Sparse Norm-C on the FEVER dataset. Both methods extract longer rationales with increasing $\pi$ that results in a decrease in agreement with sparse human rationales, while accuracy improves. However, Sparse IB consistently outperforms Sparse Norm-C in task performance.

In summary, our analysis indicates that unlike norm-minimization methods, our IB objective is able to consistently extract rationales with the specified sparsity rates, and achieves a better trade-off with accuracy. We hypothesize that optimizing the KL-divergence of the posterior $p(m|x)$ may be able to model input salience better than an implicit regularization (through $||m||_0$). The sparse prior term can learn $p(m|x)$ adaptive to different examples, while $||m||$ encourages uniform sparsity across examples.[6] This can be seen explicitly in Table 2, where the variance in sampled mask across examples is higher for our objective.

**Model Agnostic Behavior** Our approach is agnostic to choice of model architecture and word vs. sentence level rationales. We experimented with the word-level model in (DeYoung et al., 2019), where masks are learned over words instead of sentences. More details of the model architecture can be found in (DeYoung et al., 2019). The results for which are shown in Table 4

**Error Analysis** A qualitative analysis of the rationales extracted by the Sparse IB approach indi-

[6]Unlike the norm $||m||_0$, the derivative of KL-divergence term is proportional to $\log p(m|x)$

| Approach | Movies | | MultiRC | |
|---|---|---|---|---|
| | Task | IOU | Task | IOU |
| Sparse Norm-C | 91.96 | 48.9 | 64.25 | 25.7 |
| Sparse IB (Us) | **93.46** | **52.1** | **65.63** | **27.0** |

Table 4: Task and IOU F1 for our Sparse IB approach and best performing baseline on word-level rationales and BERT+LSTM model.

cates that such methods struggle when the context offers spurious—or in some cases even genuine but limited—evidence for both output labels (Figure 3). For instance, the model makes an incorrect positive prediction for the first example from the Movies sentiment dataset based on sentences that praise the *prequel* of the movie or acknowledge some critical acclaim. We also observed incorrect predictions based on shallow lexical matching (likely equating *forced* and *unforced* in the second example) and world knowledge (likely equating south *Georgia*, southeastern *United States*, and South *Florida* in the third). Overall, there is scope for improvement through better incorporation of exact lexical match, coreference propagation, and representation of pragmatics in our sentence representations.

## 6 Related Work

**Extractive Rationalization** Methods that condition predictions on their explanations are more trustworthy than post-hoc explanation techniques (Ribeiro et al., 2016; Krause et al., 2017; Alvarez-Melis and Jaakkola, 2017) and analyses of self-attention (Serrano and Smith, 2019; Jain et al., 2020). Extractive rationalization (Lei et al., 2016) is one of the most well-studied of such methods and has received increased attention with the recently released ERASER benchmark (DeYoung et al., 2019). Chang et al. (2019) and Yu et al. (2019); Chang et al. (2019) have complementary work on class-wise explanation extraction. Bastings et al. (2019) employ a reparameterizable version of the bi-modal beta distribution (instead of Bernoulli) for the binary mask. While our method has focused on unsupervised settings due to the considerable cost of obtaining reliable rationale annotations, recent work (Lehman et al., 2019) has also attempted to use direct supervision from rationale annotations for critical medical domain tasks. Finally, Latcinnik and Berant (2020) and Rajani et al. (2019) focus on generating explanations (rather than extracting them from the input). The extractive paradigm can be unfavourable for certain ERASER tasks like commonsense question answering, where the given input provides limited context for the task.

**Information Bottleneck** Information Bottleneck (IB) (Tishby et al., 1999) has recently been adapted in a number of downstream applications like parsing (Li and Eisner, 2019), extractive summarization (West et al., 2019), and image classification (Alemi et al., 2016; Zhmoginov et al., 2019). Alemi et al. (2016) and Li and Eisner (2019) use IB for optimal compression of hidden representations of images and words respectively. We are interested in compressing the number of cognitive units (like sentences) to ensure interpretability of the bottleneck representation, similar to West et al. (2019). However, while West et al. (2019) use brute-force search to optimize IB for summarization, we directly optimize a parametric variational bound on IB for rationales. IB has also been previously used for interpretability—Zhmoginov et al. (2019) use a VAE to estimate the prior distribution over $z$ for image classification. Bang et al. (2019) use IB for post-hoc explanation of sentiment classification. They do not enforce a sparse prior, and as a result, cannot guarantee that the rationale is strictly smaller than the input. Controlling sparsity to manage the accuracy-conciseness trade-off is also not possible in their model.

## 7 Conclusion

We introduce a novel sparsity-inducing objective derived from the Information Bottleneck principle to extract rationales of desired conciseness. Our approach outperforms existing norm-minimization techniques in task performance and agreement with human rationales for tasks in the ERASER benchmark. Our objective obtains a better trade off of accuracy vs. sparsity. We are also able to close the gap with models that use the full input with $< 25\%$ rationale annotations for a majority of the tasks. In future work, we would like to apply our approach on document-level and multi-document NLU tasks.

## Acknowledgments

This research was supported by ONR N00014-18-1-2826, DARPA N66001-19-2-403, ARO W911NF-16-1-0121 and NSF IIS-1252835, IIS-1562364, an Allen Distinguished Investigator Award, and the Sloan Fellowship. We thank Prof. Sreeram Kannan, Andrey Zhmoginov, and the UW NLP group for helpful conversations and comments on the work.

## References


Alexander A Alemi, Ian Fischer, Joshua V Dillon, and Kevin Murphy. 2016. Deep variational information bottleneck. *arXiv preprint arXiv:1612.00410*.

David Alvarez-Melis and Tommi Jaakkola. 2017. A causal framework for explaining the predictions of black-box sequence-to-sequence models. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 412–421.

Seojin Bang, Pengtao Xie, Heewook Lee, Wei Wu, and Eric Xing. 2019. Explaining a black-box using deep variational information bottleneck approach. *arXiv preprint arXiv:1902.06918*.

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

Iz Beltagy, Kyle Lo, and Arman Cohan. 2019. Scibert: A pretrained language model for scientific text. *arXiv preprint arXiv:1903.10676*.

Oana-Maria Camburu, Tim Rocktäschel, Thomas Lukasiewicz, and Phil Blunsom. 2018. e-snli: Natural language inference with natural language explanations. In *Advances in Neural Information Processing Systems*, pages 9539–9549.

Shiyu Chang, Yang Zhang, Mo Yu, and Tommi Jaakkola. 2019. A game theoretic approach to class-wise selective rationalization. In *Advances in Neural Information Processing Systems*, pages 10055–10065.

Christopher Clark, Kenton Lee, Ming-Wei Chang, Tom Kwiatkowski, Michael Collins, and Kristina Toutanova. 2019. Boolq: Exploring the surprising difficulty of natural yes/no questions. *arXiv preprint arXiv:1905.10044*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186.

Jay DeYoung, Sarthak Jain, Nazneen Fatema Rajani, Eric Lehman, Caiming Xiong, Richard Socher, and Byron C Wallace. 2019. Eraser: A benchmark to evaluate rationalized nlp models. *arXiv preprint arXiv:1911.03429*.

SG Fletcher and K Ponnambalam. 1996. Estimation of reservoir yield and storage distribution using moments analysis. *Journal of Hydrology(Amsterdam)*, 182(1):259–275.

Emil Julius Gumbel. 1948. *Statistical theory of extreme values and some practical applications: a series of lectures*, volume 33. US Government Printing Office.

Sarthak Jain, Sarah Wiegreffe, Yuval Pinter, and Byron C Wallace. 2020. Learning to faithfully rationalize by construction. *arXiv preprint arXiv:2005.00115*.

Eric Jang, Shixiang Gu, and Ben Poole. 2017. Categorical reparametrization with gumble-softmax. In *International Conference on Learning Representations (ICLR 2017)*. OpenReview. net.

Daniel Khashabi, Snigdha Chaturvedi, Michael Roth, Shyam Upadhyay, and Dan Roth. 2018. Looking beyond the surface: A challenge set for reading comprehension over multiple sentences. 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 252–262.

Durk P Kingma, Tim Salimans, and Max Welling. 2015. Variational dropout and the local reparameterization trick. In *Advances in neural information processing systems*, pages 2575–2583.

Josua Krause, Aritra Dasgupta, Jordan Swartz, Yindalon Aphinyanaphongs, and Enrico Bertini. 2017. A workflow for visual diagnostics of binary classifiers using instance-level explanations. In *2017 IEEE Conference on Visual Analytics Science and Technology (VAST)*, pages 162–172. IEEE.

Veronica Latcinnik and Jonathan Berant. 2020. Explaining question answering models through text generation. *arXiv preprint arXiv:2004.05569*.

Eric Lehman, Jay DeYoung, Regina Barzilay, and Byron C Wallace. 2019. Inferring which medical treatments work from reports of clinical trials. 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 3705–3717.

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

Xiang Lisa Li and Jason Eisner. 2019. Specializing word embeddings (for parsing) by information bottleneck. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2744–2754.

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

Eric Nalisnick and Padhraic Smyth. 2017. Stick-breaking variational autoencoders. In *International Conference on Learning Representations (ICLR)*.

Bo Pang and Lillian Lee. 2004. A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. In *Proceedings of the 42nd annual meeting on Association for Computational Linguistics*, page 271. Association for Computational Linguistics.

Nazneen Fatema Rajani, Bryan McCann, Caiming Xiong, and Richard Socher. 2019. Explain yourself! leveraging language models for commonsense reasoning. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4932–4942.

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.

Sofia Serrano and Noah A Smith. 2019. Is attention interpretable? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2931–2951.

James Thorne, Andreas Vlachos, Christos Christodoulopoulos, and Arpit Mittal. 2018. Fever: a large-scale dataset for fact extraction and verification. 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 809–819.

Naftali Tishby, Fernando C. Pereira, and William Bialek. 1999. The information bottleneck method. In *Proc. of the 37-th Annual Allerton Conference on Communication, Control and Computing*, pages 368–377.

Daniel S Weld and Gagan Bansal. 2019. The challenge of crafting intelligible intelligence. *Communications of the ACM*, 62(6):70–79.

Peter West, Ari Holtzman, Jan Buys, and Yejin Choi. 2019. Bottlesum: Unsupervised and self-supervised sentence summarization using the information bottleneck principle. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3743–3752.

Ronald J Williams. 1992. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning*, 8(3-4):229–256.

Mo Yu, Shiyu Chang, Yang Zhang, and Tommi Jaakkola. 2019. Rethinking cooperative rationalization: Introspective extraction and complement control. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 4085–4094.

Andrey Zhmoginov, Ian Fischer, and Mark Sandler. 2019. Information-bottleneck approach to salient region discovery. *arXiv preprint arXiv:1907.09578*.

## A Information Bottleneck Theory

We first present an overview of the variational bound on IB introduced by (Alemi et al., 2016) and then derive a modified version amenable to interpretability.

### A.1 Variational Information Bottleneck (Alemi et al. (2016))

The objective is to parameterize the information bottleneck objective $L_{IB} = I(X, Z) - \beta I(Z, Y)$ using neural models and use SGD to optimize. Consider the joint distribution: $p(X, Y, Z) = p(Z|X, Y)p(Y|X)p(X) = p(Z|X)p(Y|X)p(X)$ under the Markov chain $Y \leftrightarrow X \leftrightarrow Z$. As mutual information is hard to compute, the following bounds are derived on both MI terms:

First Term:

$$I(Z, X) := \mathbb{E}_x \left[ \mathop{\mathbb{E}}_{z \sim p_\theta(z|x)} \left[ \log \frac{p_\theta(z|x)}{p(z)} \right] \right]$$

where,

$$p(z) := \int dx p_\theta(z|x)p(x)$$

This marginal is intractable. Let $r(z)$ be a variational approximation to this marginal. Since $\text{KL}[p(z), r(z)] \geq 0$,

$$I(Z, X) \leq \mathbb{E}_x \left[ \mathop{\mathbb{E}}_{z \sim p_\theta(z|x)} \left[ \log \frac{p_\theta(z|x)}{r(z)} \right] \right]$$

If $p_\theta(z|x)$ and $r(z)$ are of a form that KL divergence can be analytically computed, we get:

$$I(Z, X) \leq \mathbb{E}_x \left[ \text{KL}[p_\theta(z|x), r(z)] \right]$$

Typically, the distributions $p_\theta(z|x)$ and $r(z)$ are instantiated as multivariate Normal distributions to analytically compute the KL-divergence term.

$$r(z) = \mathcal{N}(z|0, I), \;\; p(z|x) = \mathcal{N}(z|\mu(x), \Sigma(x));$$

where $\mu$ is a neural network which outputs the K-dimensional mean of z and $\Sigma$ outputs the $K \times K$ covariance matrix $\Sigma$. This also allows us to reparameterize samples drawn from $p_\theta(z|x)$.

Second Term:

$$I(Z, Y) := \mathop{\mathbb{E}}_{y, z \sim p_\theta} \left[ \log \frac{p(y|z)}{p(y)} \right]$$

where,

$$p(y|z) := \int dx \frac{p(y|x)p(z|x)p(x)}{p(z)}$$

Again, as this is intractable, $q_\phi(y|z)$ is used as a variational approximation to $p(y|z)$ and is instantiated as a transformer model with its own set of parameters $\phi$. As Kullback Leibler divergence is always positive:

$$\text{KL}[p(y|z), q_\phi(y|z)] \geq 0 \rightarrow$$

$$I(Z, Y) \geq \mathop{\mathbb{E}}_{y, z \sim p_\theta} \left[ \log \frac{q_\phi(y|z)}{p(y)} \right]$$

The term $p(y)$ can be dropped as it is constant with respect to parameters $\phi$. Thus, we minimize $\mathbb{E}_{y,z \sim p_\theta}[-\log q_\phi(y|z)]$ Thus the IB objective is bounded by the loss function:

$$L_{vib} \geq \mathbb{E}_{y,z \sim p_\theta}[-\log q_\phi(y|z)] + \beta \text{KL}[p_\theta(z|x), r(z)]$$

### A.2 Deriving the Sparse Prior Objective

The latent space learned in Appendix A.1 is not easy to interpret. Instead we consider a masked representation of the form $z = m \odot x$, where $m_j \in \{0, 1\}$ is a binary mask sampled from a distribution $p_\theta(m_j|x) = \text{Bernoulli}(\theta_j(x))$. This is an adaptive masking strategy, defined by data-driven relevance estimators $\theta_j(x)$. The distributions over $x$ and $m$ induce a distribution on $z = m \odot x$ defined by the conditionals

$$p_\theta(z_j|x) = (1 - \theta_j(x))\delta(z_j) + \theta_j(x)\delta(z_j - x_j).$$

Our prior, based on human annotations, is that rationale needed for a prediction is sparse; we encode this prior as a distribution over masks $r(m_j) = \text{Bernoulli}(\pi)$. The prior also induces a distribution on $z = m \odot x$ given by

$$r(z_j|x) = (1 - \pi)\delta(z_j) + \pi\delta(z_j - x_j).$$

We want to enforce a constraint $p_\theta(z_j) = r(z_j)$; i.e. that the marginal distribution $p_\theta(z_j) = \int p_\theta(z_j|x)p(x)\, dx$ matches our prior $r(z_j)$. This is difficult to do directly, but as in Appendix A.1, we can construct an upper bound the mutual information between $x$ and $z$:

$$I(Z, X) \leq \mathop{\mathbb{E}}_{x \sim p} \left[ \text{KL}[p_\theta(z|x), r(z)] \right].$$

The inequality is tight if $r(z) = p_\theta(z)$. By optimizing to minimize mutual information $I(Z, X)$, we

| Hyperparameter | Movie | FEVER | MultiRC | BoolQ | Evidence Inference | BEER |
|---|---|---|---|---|---|---|
| NS | 36 | 10 | 15 | 25 | 20 | 10 |
| $\pi$ (Sparsity threshold (%)) | .40 | .20 | .25 | .20 | .20 | .20 |
| $\gamma$ (weight on SR) | 0.5 | 0.05 | 1.00E-04 | 0.01 | 0.001 | 0.01 |

Table 5: Hyperparameters used to report results.

| Approach | FEVER | | MultiRC | | Movies | | BoolQ | | Evidence | |
|---|---|---|---|---|---|---|---|---|---|---|
| | Task | IOU | Task | IOU | Task | IOU | Task | IOU | Task | IOU |
| Full | 90.54 | - | 68.18 | - | 88.0 | - | 63.16 | - | 47.51 | - |
| Gold | 92.52 | - | 78.20 | - | 1.0 | - | 71.65 | - | 85.39 | - |
| No Sparsity | 83.01 | 35.50 | 59.17 | 22.42 | 81.46 | 20.63 | 61.82 | 10.39 | 47.51 | 9.87 |
| Sparse Norm | 84.30 | 45.44 | 58.40 | 20.41 | 79.35 | 19.23 | 59.04 | 12.40 | 44.52 | 9.4 |
| Sparse Norm-C | 84.42 | 44.90 | 60.77 | 23.25 | 82.43 | 18.91 | 62.24 | 09.72 | 48.97 | 09.40 |
| Sparse IB | **85.64** | **45.46** | **61.11** | **25.55** | **86.50** | **22.33** | **63.07** | **16.63** | **49.09** | **11.09** |

Table 6: Final results of our unsupervised models on ERASER Dev Set

will implicitly learn parameters $\theta$ that approximate the desired constraint on the marginal.

In contrast to Alemi et al. (2016), our prior $r(z)$ has no parameters; rather than using an expressive model $r(z)$ to approximate the $p_\theta(z)$, we instead use the fixed prior $r(z)$ to force the learned conditionals $p_\theta(z|x)$ to assume a form such that the marginal $p_\theta(z)$ approximately matches the marginal of the prior, $\pi$. Average mask sparsity values in Table 2 corroborate this.

By a limiting argument, we can compute the divergence between $p_\theta(z|x)$ and $r(z)$:

$$\begin{aligned}
&\mathrm{KL}(p_\theta(z_j|x), r(z_j)) \\
&\quad = (1-\theta_j(x)) \int \delta(z_j) \log \frac{p_\theta(z_j|x)}{r(z_j)}\, dz_j \\
&\quad + \theta_j(x) \int \delta(z_j - x_j) \log \frac{p_\theta(z_j|x)}{r(z_j)}\, dz_j \\
&\quad = (1-\theta_j(x)) \log \frac{1-\theta_j(x)}{1-\pi} + \theta_j(x) \log \frac{\theta_j(x)}{\pi p(x)} \\
&\quad = \mathrm{KL}(p_\theta(m_j|x), r(m_j)) - \theta_j(x) \log p(x).
\end{aligned}$$

The term $\mathrm{KL}[p_\theta(m_j|x), r(m_j)]$ is a divergence between two Bernoulli distributions and has a simple closed form. If $\theta_j(x)$ and $\log p(x)$ are uncorrelated then

$$\mathop{\mathbb{E}}_{x\sim q}[-\theta_j(x) \log p(x)] = \pi H(X).$$

The term $\pi H(X)$ is constant with respect to the parameters $\theta$ and can be dropped.

We use the same, standard cross-entropy bound discussed in Appendix A.1 to estimate $I(Z, Y)$, leading us to our variational bound on IB with interpretability constraints

$$\begin{aligned}
L_{IVIB} = &\mathbf{E}_{m\sim p(m|x)}[-\log q(y|m \odot x)] \\
&+ \beta \sum_j KL[p_\theta(m_j|x)||r(m_j)].
\end{aligned}$$

## B Experimental Details

### B.1 Data Processing

The train, test and validation splits are the same as used in the ERASER benchmark (DeYoung et al., 2019) and for the Beer Advocate dataset (Bastings et al., 2019). In order to batch operations, we process the data so that each example has at most NS sentences. NS is fixed based on the average number of sentences in the development set of the respective task (see Table 5). Some dataset specific processing details are highlighted below:

**FEVER**: ERASER adapts the original fact verification task as a binary classification of whether the given evidence supports or refutes a given claim.

**MultiRC**: The reading comprehension task with multiple correct answers is modified into a binary classification task for ERASER, where each (rationale, question, answer) triplet has a true/false label.

**BoolQ**: A Boolean (yes/no) question answering dataset over Wikipedia articles. Since most documents are considerably longer than BERT's maximum context window length of 512 tokens (3.3K

tokens on average), we use a sliding window to select a single document *span* that has the maximum TF-IDF score against the question.

**Evidence Inference**: A three-way classification task over full-text scientific articles for inferring whether a given medical intervention is reported to either *significantly increase*, *significantly decrease*, or have *no significant effect* on a specified outcome compared to a comparator of interest. We again apply the TF-IDF heuristic as the average number of tokens is a document is 4.6K.

**BEER**: The Beer Advocate regression task for predicting 0-5 star ratings for multiple aspects like appearance, smell, and taste based on reviews. We report on the appearance aspect.

### B.2 Modeling

For question answering tasks in ERASER. $s$ and $x$ are encoded together in the sequence $s\texttt{[SEP]}x$ while assuming that $s$ is fully unmasked i.e. $p_\theta(m_s|x) = 1$. Once again, the sequence $s\texttt{[SEP]}m \odot x$ is used if query $s$ is available, i.e., we assume no masking over $s$ as it is assumed to be essential to predict $y$.

**Semi-supervised:** Whenever train loss is not available, only task loss is used. Evaluation is still done based on $\pi\%$ sentences, to fairly compare with unsupervised models.

### B.3 Hyperparameters

We use a sequence length of 512, batch size of 16 [7] and Adam optimizer with a learning rate of 5e-5. We do not use warm-up or weight decay. We run all model for 20 epochs and set patience to 10 (over iterations). Hyper-parameter tuning is done on the validation set for the rationale performance metric (IOU F1[8]) on the development sets for ERASER tasks and on the test set for BEER (only test set contains rationale annotations). We tune the value of $\pi \in \{0.05, 0.1, 0.15, ...0.50\}$. We found that Sparse IB approach is not as sensitive to the parameter $\beta$ and fix it to 1 to simplify experimental design. For baselines, we tune the values of the Lagrangian multipliers, $\lambda \in \{1e\text{-}4, 5e\text{-}4, 1e\text{-}3, \ldots, 1\}$ as norm-based techniques are more sensitive to $\lambda$. The value of the $\gamma$ hyperparameter in the semi-supervised setup was set to 1.0 to simplify design. Instead of explicitely tuning or annealing the Gumbel softmax parameter, we fix it to 0.7 across all our experiments (including baselines). Hyperparameters for each dataset used for the final results are presented in Table 5.[9]

## C Analysis

**Learning the Value of $\pi$** Instead of tuning the value of $\pi$, we can alternately learn an appropriate value by allowing $\pi$ to be a learnable parameter in our implementation. In our experiments (see Table 7, we found that that the norm-minimization completely degenerates and learns a very high value of $\pi$, as the norm-loss in Equation 6 (Section 4.3) can still be minimized if both $||m||$ and $\pi$ are driven close to 1.0. In our case, since $pi$ is now a learnable parameter, we have to minimize the following objective.

$$L_{IVIB} = \mathbf{E}_{m\sim p_\theta(m|x)}[-\log q_\phi(y|m \odot x)] + \beta \sum_j KL[p_\theta(m_j|x)||r(m_j)] + \pi H(x) \quad (7)$$

The caveat here is that it requires another hyperparameter, namely the constant $H(x) = \lambda$ [10]. This is not unlike Sparse Normor Sparse Norm-Cwhere sparsity is controlled through the hyperparameter $\lambda$. In Table 7, we compare the Sparse IBobjective with Equation 7 for Movies and FEVER. We find that optimizing Equation 7 actually allows us to control the trade-off because of the presence of the term $\pi H(x)$ that enforces a smaller value for $\pi$. The learned value of $\pi$ is close the tuned value in Table 5, thus we choose to report our main results across all models on tuned $\pi$.

**A More Expressive Distribution** Bastings et al. (2019) compare against the best-known previous work on norm regularization Lei et al. (2016) by exploring the bi-modal Kumaraswamy distribution (Fletcher and Ponnambalam, 1996) to replace the Bernoulli distribution. This more expressive distribution may be able to complement our approach, as KL-divergence for it can be analytically computed (Nalisnick and Smyth, 2017) (Appendix C).

[7] We used 2 GeForce GTX TITAN X GPUs and Cuda 10.1

[8] Calculated as per the definition in https://github.com/jayded/eraserbenchmark/blob/master/rationale_benchmark/metrics.py for threshold 0.1

[9] We observed some variation (<0.50 F1) in results across across GPUs, well within the difference observed between Sparse IB and baselines.

[10] We could alternately estimate this using a VAE, as done in Zhmoginov et al. (2019)

| Approach | Movies | | | Fever | | |
|---|---|---|---|---|---|---|
| | Task F1 | IOU F1 | Sparsity | Task F1 | IOU F1 | Sparsity |
| Sparse Norm-Cwith learned $\pi$ | 89.86 | 24.18 | 0.99 | 89.0 | 36.2 | 0.98 |
| Sparse IB | 91.0 | 24.18 | 0.98 | 88.50 | 36.2 | 0.96 |
| Sparse IBwith learned $\pi$ | 86.97 | 25.63 | 0.45 | 85.64 | 45.71 | 0.14 |

Table 7: Evaluation of learnable $\pi$. Results on Dev set

| Distribution/Approach | Movies | | Fever | |
|---|---|---|---|---|
| | Task | IOU | Task | IOU |
| Bernoulli (Sparse Norm-C) | 79.4 | 18.3 | 83.3 | 44.9 |
| Bernoulli Sparse IB | **81.5** | **21.8** | **84.7** | **45.5** |
| Kuma Sparse Norm-C | 81.8 | 21.0 | 84.9 | 43.0 |
| Kuma Sparse IB | **83.4** | **21.5** | **85.6** | **45.5** |

Table 8: Results on the Kumaraswamy distribution from (Bastings et al., 2019) on Dev set

The KL divergence between the Kumaraswamy and Beta distribution can be analytically computed, as done in this work (Nalisnick and Smyth, 2017). In Table 8, we show results on Movies and FEVER datasets for this distribution, comparing Sparse IBagainst the Sparse Norm-Cbaseline. We find that the superior performance of the KL-divergence loss term persists.