An Information Bottleneck Approach for Controlling Conciseness in Rationale Extraction

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

Abstract

Decisions of complex language understanding models can be rationalized by limiting their inputs to a relevant subsequence of the original text. A rationale should be as concise as possible without significantly degrading task performance, but this balance can be difficult to achieve in practice. In this paper, we show that it is possible to better manage this trade-off by optimizing a bound on the Information Bottleneck (IB) objective. Our fully unsupervised approach jointly learns an explainer that predicts sparse binary masks over sentences, and an end-task predictor that considers only the extracted rationale. Using IB, we derive a learning objective that allows direct control of mask sparsity levels through a tunable sparse prior. Experiments on ERASER benchmark tasks demonstrate significant gains over norm-minimization techniques 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) closes the gap with a model that uses the full input.†

1 Introduction

Rationales that select the most relevant parts of an input text can help explain model decisions for a range of language understanding tasks (Lei et al., 2016; DeYoung et al., 2019). Models can be faithful to a rationale by only using the selected text as input for end-task prediction. However, there is almost always a trade-off between interpretable models that 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).

Figure 1: Our Information Bottleneck-based approach extracts concise rationales as a compressed intermediate bottleneck representation. In this fact verification example, the model uses a Boolean mask to select 40% of the input sentences (boldfaced) based on the specified sparse prior 0.4, and predicts the label (refutes) conditioned only on the masked input.

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). As we will see in Section 5, it is 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. 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 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 bottle-
neck) 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 interpretability.

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 different 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 methods, our KL-divergence objective is able to build an interpretable 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:

\[ L_{IB} = I(X, Z) - \beta I(Z, Y) \]

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:\(^2\)

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

where \( q_{\phi}(y|z) \) is a parametric approximation to the true likelihood \( p(y|z) \); \( r(z) \), the prior probability of \( z \), is an approximation to \( p(z) \); 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\(^3\) between the posterior distribution \( p_{\theta}(z|x) \) that depends on \( x \) and a prior distribution \( r(z) \) that is

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

3.\(^{\text{To analytically compute the KL-divergence term, the posterior and prior distributions over } z \text{ are typically K-dimensional multivariate normal distributions. Compression is achieved by setting } K << D, \text{ the input dimension of } X.} \)

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 \( x = (x_1, x_2, \ldots, x_n) \). \( y \) 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 \( x \setminus z \) (DeYoung et al., 2019).

2. \( z \) must be compact yet sufficient, 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 \( y \). We elaborate on how sufficiency is attained using Information Bottleneck in the following section.

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:

\[ L_{IB} = I(X, Z) - \beta I(Z, Y) \]

Equation 1:

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

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:

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

where \( q_{\phi}(y|z) \) is a parametric approximation to the true likelihood \( p(y|z) \); \( r(z) \), the prior probability of \( z \), is an approximation to \( p(z) \); 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\(^3\) between the posterior distribution \( p_{\theta}(z|x) \) that depends on \( x \) and a prior distribution \( r(z) \) that is
We consider an interpretable latent representation $\delta$. The learned bottleneck representation via Equation 2, is often not human-interpretable. Hence, we obtain the following variation, in contrast to compression via dimensionality reduction of the input via sparsity in the latent representation, in contrast to compression via dimensionality reduction (Alemi et al., 2016).

For the masked representation $z = m \odot x$, we can decompose we can decompose $KL(p_m(z_j|x), r(z_j))$ as:

$$KL(p_m(z_j|x), r(z_j)) = KL(p_m(m_j|x), r(m_j)) + \pi H(x)$$

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

$$L_{IVIB} = E_{m \sim p(m|x)}[-\log q(y|m \odot x)] + \beta \sum_j KL[p_m(m_j|x)||r(m_j)]$$

where $\pi$ is the expected cross-entropy term for the task which can be computed by drawing samples $m \sim p_m(m|x)$. The information-loss term encourages the mask $m$ to be independent of $x$ by reducing the KL divergence of its posterior $p_m(m|x)$ 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 $z = x$ and the rationale is no longer concise. In the following section, we present a simple way to avoid this degenerate case in practice.

### 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 $r(m)$ 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 is $\pi = 0.10$, setting $r(m_j) = \pi = 0.1 \forall j$. We refer to this prior probability hyperparameter as the sparsity threshold $\pi$. $\pi$ can be estimated as the expected sparsity of the mask from expert annotations. If such a statistic is not available, it can be explicitly tuned using $0 \leq \pi \leq 0.5$, for the desired trade-off between end task performance and rationale length.

Consequently, we can control the amount of sparsity in the mask that is eventually sampled from the learned posterior distribution $p_m(m|x)$ by appropriately setting the prior belief $r(m) = \pi$ in Equation 3. Since $\pi$ is restricted to be small, the sampled mask is generally sparse. As a result, the intermediate representation $z = x \odot m$, which is our human-interpretable rationale, is guaranteed to be a subsequence of $x$ and reduce $I(Z, X)$. We refer to this training objective as the sparse prior (Sparse IB) method in our experiments.

### 3 Model

#### 3.1 Architecture

Our model (Figure 2) consists of an explainer which extracts rationales from the input, and an end-task predictor which predicts the output based on the explainer rationales. In our experiments, both the explainer and the predictor are Transformers with BERT pretraining (Devlin et al., 2019).

**Explainer $p_m(z|x)$:** 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 \( x = (x_1, x_2, \ldots, x_n) \) where \( x_j \) is obtained by concatenating the contextualized representations of the first and last tokens in sentence \( x_j \). When an optional query sequence \( s \) is available,\(^4\) and \( x \) are encoded together in the sequence \( s \{ \text{[SEP]} \} x \) while assuming that \( s \) is fully unmasked i.e. \( p_\theta(m_s|x) = 1 \). 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. The mask \( m \in \{0, 1\}^n \) is constructed by independently sampling each \( m_j \) from \( p(m_j|x) \). We define \( z \) as the rationale representation \( z = m \odot x \), an element-wise dot product between \( m_j \) and the corresponding sentence representation \( x_j \).

**End-task Predictor \( q_\phi(y|z) \):** The end-task predictor applies the mask \( m \) from the explainer to the input sequence\(^5\) \( x \) to predict the output variable \( y \). The same 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).

\(^4\)For question answering tasks in ERASER.

\(^5\)Once again, the sequence \( s \{ \text{[SEP]} \} m \odot x \) is used if query \( s \) is available, i.e., we assume no masking over \( s \) as it is assumed to be essential to predict \( y \).

### 3.2 Reparameterization and Training

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 trick (Jang et al., 2017) for categorical (here, binary) distributions to reparameterize the Bernoulli variables \( m_j \). A random perturbation \( e_j \sim U(0, 1) \) is added to the log probability (logit), \( \log p_\theta(m_j|x) \). The reparameterized binary variable \( m_j^* \) is generated as follows:

\[
g_j = -\log (\log (e_j))
\]

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

where \( \sigma \) is the Sigmoid function, \( \tau \) is a hyperparameter for the temperature of the Gumbel-Softmax function, and \( g_j \) is a 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.

**Inference:** During inference, we extract the top \( \pi \% \) sentences, 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 judge how well the model fills the budget with the best rationales.

### 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.
We evaluate performance on five text classification tasks from the ERASER benchmark (DeYoung et al., 2019) and one regression task used in previous work (Lei et al., 2016). All these datasets have sentence-level rationale annotations for validation and test sets. Additionally, the ERASER tasks contain rationale annotations for the training set, which we only use for our semi-supervised experiments.

**Movies** (Pang and Lee, 2004): Sentiment classification of movie reviews from IMDb.

**FEVER** (Thorne et al., 2018): A fact extraction and verification task adapted in ERASER as a binary classification of the given evidence supporting or refuting a given claim.

**MultiRC** (Khashabi et al., 2018): A reading comprehension task with multiple correct answers modified into a binary classification task for ERASER, where each (rationale, question, answer) triplet has a true/false label.

**BoolQ** (Clark et al., 2019): 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, we use a sliding window to select a single document span that has the maximum TF-IDF score against the question.

**Evidence Inference** (Lehman et al., 2019): 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.

**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. We report on the appearance aspect.

### 4.2 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. For task accuracy, we report weighted F1 for classification tasks, and the mean square error for the BeerAdvocate regression task.

### 4.3 Baselines

**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:

$$L_{SL0} = E_{m \sim p(m|x)} \left[-\log q(y|z)\right] + \lambda||m||$$  

where $\lambda$ is the weight on the norm.

**Controlled Norm Minimization (Sparse Norm-C):** For fair comparison against our approach for controlled sparsity, we can modify Equation 4 to ensure that the norm of $m$ is not penalized when it drops below the threshold $\pi$.

$$L_{SL0-C} = E_{m \sim p(m|x)} \left[-\log q(y|z)\right] + \lambda \max \left(0, \sum_j ||m_j|| - \pi\right)$$

Explicit control over sparsity in the mask $m$ through the tunable prior probability $\pi$ naturally emerges from IB theory in our Sparse IB approach, as opposed to the modification adopted in norm-based regularization (Equation 5).

**No Sparsity (Task Only):** This method only optimizes for the end-task performance without any sparsity-inducing loss term, and serves as a common baseline for evaluating the effect of sparsity inducing objectives in Sparse IB, Sparse Norm, and Sparse Norm-C.

In addition to the above baselines, we also compare to models that don’t predict rationales:

For input example $x = (x_1, x_2, \ldots, x_n)$ and a gold mask $\hat{m} = (\hat{m}_1, \hat{m}_2, \ldots, \hat{m}_n)$, we use the following semi-supervised objective:

$$L_{semi} = E_{m \sim p_0(m|x)} \left[-\log q(y|m \odot x)\right] + \gamma \sum_j -\hat{m}_j \log p(m_j|x)$$

We set $\gamma = 1$ to simplify experiments. For examples where the rationale supervision is not available, we only consider the task loss.

### 4 Experimental Setup

#### 4.1 End Tasks

We evaluate performance on five text classification tasks from the ERASER benchmark (DeYoung et al., 2019) and one regression task used in previous work (Lei et al., 2016). All these datasets have sentence-level rationale annotations for validation and test sets. Additionally, the ERASER tasks contain rationale annotations for the training set, which we only use for our semi-supervised experiments.

**Movies** (Pang and Lee, 2004): Sentiment classification of movie reviews from IMDb.

**FEVER** (Thorne et al., 2018): A fact extraction and verification task adapted in ERASER as a binary classification of the given evidence supporting or refuting a given claim.

**MultiRC** (Khashabi et al., 2018): A reading comprehension task with multiple correct answers modified into a binary classification task for ERASER, where each (rationale, question, answer) triplet has a true/false label.

**BoolQ** (Clark et al., 2019): 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, we use a sliding window to select a single document span that has the maximum TF-IDF score against the question.

**Evidence Inference** (Lehman et al., 2019): 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.

**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. We report on the appearance aspect.

### 4.2 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. For task accuracy, we report weighted F1 for classification tasks, and the mean square error for the BeerAdvocate regression task.

### 4.3 Baselines

**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:

$$L_{SL0} = E_{m \sim p(m|x)} \left[-\log q(y|z)\right] + \lambda||m||$$  

where $\lambda$ is the weight on the norm.

**Controlled Norm Minimization (Sparse Norm-C):** For fair comparison against our approach for controlled sparsity, we can modify Equation 4 to ensure that the norm of $m$ is not penalized when it drops below the threshold $\pi$.

$$L_{SL0-C} = E_{m \sim p(m|x)} \left[-\log q(y|z)\right] + \lambda \max \left(0, \sum_j ||m_j|| - \pi\right)$$

Explicit control over sparsity in the mask $m$ through the tunable prior probability $\pi$ naturally emerges from IB theory in our Sparse IB approach, as opposed to the modification adopted in norm-based regularization (Equation 5).

**No Sparsity (Task Only):** This method only optimizes for the end-task performance without any sparsity-inducing loss term, and serves as a common baseline for evaluating the effect of sparsity inducing objectives in Sparse IB, Sparse Norm, and Sparse Norm-C.

In addition to the above baselines, we also compare to models that don’t predict rationales:
Table 1: Task and Rationale IOU F1 for our Sparse IB approach and baselines (Section 4.3) on test sets. Pipeline refers to the Bert-to-Bert method reported in DeYoung et al. (2019), while we use 25% training data in our semi-supervised setting (Section 3.3). We report MSE for BeerAdvocate, hence lower is better. BeerAdvocate has no training rationales. Gold IOU is 100.0. Validation set results can be found in Table 5 in the Appendix.

Full Context (Full): This method uses the entire context to make prediction, and allows us to estimate the loss in performance as a result of our interpretable hard attention model that only uses $\pi\%$ of the input.

Gold Rationale (Gold): For datasets with human-annotated rationales available at training time, we train a model that uses these rationale annotations for training and inference to estimate an upper-bound that can be achieved on task and rationale performance metrics with transformers.

4.4 Implementation Details

We use BERT-base with a maximum context-length of 512 to instantiate the explainer and end-task predictor. Models are tuned based on their performance on the rationale IOU F1 as it is available for all the datasets considered in this work. When rationale IOU F1 is not available, the sparsity-accuracy trade-off (Figure 4) can be used to determine an operation point. We tune the prior probability/threshold $\pi \in (0, 0.5)$ to ensure strictly concise rationales. For our Sparse IB approach, we observe less sensitivity to hyperparameter $\beta$ and set it to 1 to simplify experimental design. For baselines, we tune the values of the Lagrangian multipliers, $\lambda \in \{1e-4, 5e-4, 1e-3, \ldots, 1\}$ as norm-based techniques are more sensitive to $\lambda$. More details about the hyperparameters and model selection are presented in Appendix B.

5 Results

Table 1 compares our Sparse IB approach against baselines described in Section 4.3. Our Sparse IB approach 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 relative improvements ranging from 5 to 80% over the better performing norm-minimization variant Sparse Norm-C. Sparse IB also attains task performance within $0.5 - 10\%$ of the full-context model (row 1), despite using $< 40\%$ of the input sentences on average. All unsupervised approaches still obtain a lower IOU F1 compared to the full-context model for the Movies and MultiRC datasets, primarily due to their considerably lower precision on these benchmarks.

Our results also highlight the importance of explicit controlled sparsity inducing terms as essential inductive biases for improved task performance and rationale agreement. Specifically, sparsity-inducing methods consistently outperform the Task Only-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 observe a positive correlation between task performance and 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 ratio-
Figure 3: Semi-supervised experiments showing the task performance for varying proportions of rationale annotation supervision on the MultiRC (left) and FEVER (right) 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. SIB is Sparse IB and SN-C is Sparse Norm-C.

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.

5.1 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 sparsity achieved by the explainer during inference by computing the average number of one entries in the input.

---

6Selected document spans have gold rationales 51.8% of the time
put mask $m$ over sentences (the hamming weight) for 100 runs. Our Sparse IB-approach 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 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 value of $\pi$ that results in a decrease in agreement with sparse human rationales, while model accuracy improves. However, Sparse IB consistently outperforms Sparse Norm-C in terms of task performance.

In summary, our analysis indicates that unlike norm-minimization methods, our KL-divergence 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 $|m|_0$ encourages uniform sparsity across examples. This can be seen explicitly in Table 2, where the variance in sampled mask across examples is higher for our objective.

**Error Analysis** A qualitative analysis of the rationales extracted by the Sparse IB approach indicates 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: (a) praise the prequel of the movie, (b) still acknowledge some critical acclaim, and (c) sarcastically describe the movie as magical. 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

Interpretability Previous work on explaining model predictions can be broadly categorized into post hoc explanation methods and methods that integrate explanations into the model architecture. Post hoc explanation techniques (Ribeiro et al., 2016; Krause et al., 2017; Alvarez-Melis and Jaakkola, 2017) typically approximate complex decision boundaries with locally linear or low complexity models. While post hoc explanations often have the advantage of being simpler, they are not faithful by construction.

On the other hand, methods that condition predictions on their explanations can be more trustworthy. Extractive rationalization (Lei et al., 2016) is one of the most well-studied of such methods in NLP, and has received increased attention with the recently released ERASER benchmark (DeYoung et al., 2019). Building on Lei et al. (2016), Chang et al. (2019) and Yu et al. (2019) consider benefits like class-wise explanation extraction while Chang et al. (2020) explore invariance to domain shift. Bastings et al. (2019) employ a reparameterizable version of the bi-modal beta distribution (instead of Bernoulli) for the binary mask. This more expressive distribution may be able to complement our approach, as KL-divergence for it can be analytically computed (Nalisnick and Smyth, 2017).

IB has also been previously used for interpretability—Zhmoginov et al. (2019) use a VAE to estimate the prior and posterior distributions over the intermediate representation $z$ for image classification. Bang et al. (2019) use IB for post-hoc explanation for sentiment classification. They do not enforce a sparse prior, and as a result, cannot guarantee that the rationale is strictly smaller than the input. This also means controlled sparsity, which we have shown to be crucial for task performance and rationale extraction, is harder to achieve in their model.

7 Conclusion

We propose a new sparsity 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 annotations for rationales for tasks in the ERASER benchmark. The sparse prior objective also allows for a straight-forward and accurate control of the amount of sparsity desired in the rationales. We also obtain better a trade off of accuracy vs. sparsity using our objective. We are 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 explore the application of our approach on longer contexts and tasks such as document-level QA.
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.

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

Shiyu Chang, Yang Zhang, Mo Yu, and Tommi S Jaakkola. 2020. Invariant rationalization. arXiv preprint arXiv:2003.09772.

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.

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

Eric Jang, Shixiang Gu, and Ben Poole. 2017. Categorical reparameterization 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.

James Thorne, Andreas Vlachos, Christos Christodouloupolous, 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:

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

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 $\text{KL}[p(z), r(z)] \geq 0,$

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

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.

$$r(z) = N(z|0, I), \quad p(z|x) = N(z|\mu(x), \Sigma(x))$$

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:

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

\[
\begin{align*}
\text{KL}[p(y|z), q_\phi(y|z)] &\geq 0 \\
I(Z, Y) &\geq \mathbb{E}_{y, z \sim p_\theta} \left[ \log \frac{q_\phi(y|z)}{p(y)} \right]
\end{align*}
\]

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:

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

### 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 \( z \):

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

The inequality is tight if \( r(z) = p_\theta(z) \). By optimizing to minimize mutual information \( I(Z, X) \), we 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. Average mask sparsity values in Table 2 corroborate this.

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

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

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

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

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

\[
L_{\text{IV}IB} = \mathbb{E}_{m \sim p(m|x)} [-\log q(y|m \odot x)] + \beta \sum_j \text{KL}[p_\theta(m_j|x) || r(m_j)].
\]

### B Experimental Details

**Hyperparameters** We use a sequence length of 512, batch size 16 and Adam optimizer with a learning rate of 5e-5. We do not use warmup or weight decay. Hyper-parameter tuning is done on the validation set for the rationale performance metric (IOU F1) for ERASER tasks and on the test set for BEER (only test set contains rationale annotations). Instead of explicitly tuning or annealing the Gumbel softmax parameter, we fix it to 0.7 across all our
| Hyperparameter         | Movie | FEVER | MultiRC | BoolQ | Evidence Inference | BEER |
|------------------------|-------|-------|---------|-------|--------------------|------|
| Num. Sentences         | 36    | 10    | 15      | 25    | 20                 | 10   |
| π (Sparsity threshold (%)) | 40    | 10    | 25      | 20    | 20                 | 20   |
| γ (weight on SR)       | 0.5   | 0.05  | 1.00E-04| 0.01  | 0.001              | 0.01 |

Table 4: Hyperparameters used to report results

| Approach          | FEVER Task | FEVER IOU | MultiRC Task | MultiRC IOU | Movies Task | Movies IOU | BoolQ Task | BoolQ IOU | Evidence Task | Evidence IOU |
|-------------------|------------|-----------|--------------|-------------|-------------|------------|------------|------------|----------------|--------------|
| Full              | 90.54      | -         | 68.18        | -           | 88.0        | -          | 63.16      | -          | 47.51          | -            |
| Gold              | 92.52      | -         | 78.20        | -           | 1.0         | -          | 71.65      | -          | 85.39          | -            |
| Task Only         | 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      | 49.67          | 09.40        |
| Sparse IB         | 85.64      | 45.46     | 61.11        | 25.55       | 86.50       | 22.33      | 62.07      | 16.63      | 49.09          | 11.09        |

Table 5: Final results of our unsupervised models on ERASER Dev Set

experiments. We found that Sparse IB approach is not as sensitive to the parameter $\beta$ and fix it to 1 to simplify experimental design. Hyperparameters for each dataset used for the final results are presented in Table 4.

Data Processing For ERASER tasks, we use the preprocessed training and validation sets from DeYoung et al. (2019) and for BEER, we use data preprocessed for the appearance aspect in Bastings et al. (2019). For BoolQ and Evidence Inference, we use a sliding window of 20 sentences (with step 5) over the document to find the span that has maximum TF-IDF overlap with the query.