LOW-ENTROPY FEATURES HURT OUT-OF-DISTRIBUTION PERFORMANCE

Anonymous authors
Paper under double-blind review

ABSTRACT

We study the relationship between the entropy of intermediate representations and a model's robustness to distributional shift. We train two feed-forward networks end-to-end separated by a discrete n-bit channel on an unsupervised contrastive learning task. Different *masking strategies* are implemented that remove a proportion p_{mask} of low-entropy bits, high-entropy bits, or random bits, and the effects on performance are compared to the baseline accuracy with no mask. When testing in-distribution (InD) we find that the removal of bits via any strategy leads to an *increase* in performance, when masking out a relatively low p_{mask}. We hypothesize that the entropy of a bit serves as a guide to its usefulness out-of-distribution (OOD). Through experiment on three OOD datasets we demonstrate that the removal of low-entropy bits can notably benefit OOD performance. Conversely, we show that top-entropy masking disproportionately harms performance both InD and OOD.

1 INTRODUCTION

The key challenge that we seek to address is that of identifying features in a model's intermediate representation that are more likely to be robust to distributional shift. Our approach starts from aiming to be robust to a class of distributional shifts where 'abstract' features in the learning domain tend to 'degrade less' than more specific features. We measure 'abstract' versus 'specific' by means of the entropy of a feature. To give intuition to this motivation, consider some computer-vision task where representing the feature 'blue sky' is useful, and 50% of the training images contain blue sky. As such, the intermediate representation of this feature has maximum entropy. On the other hand, suppose that 1% of the images contain a rare species of tree indigenous to the region where the photos where captured, e.g the 'Socotra dragon tree'. Representations of this feature would have very low entropy, but they would be of occasional use in the training distribution. Thus, given a distributional shift such as moving to another region, the 'blue sky' feature would remain useful, but the 'Socotra dragon tree' would not. Put differently, we hypothesize that the low-entropy features tend to be more niche and domain-specific, and so will become irrelevant or even harmful out-of-distribution (OOD).

The main contributions of this paper are: firstly, demonstrating that the removal of low-entropy representations via the masking of learned discrete bits can notably *improve* OOD performance. Secondly, showing that the removal of high-entropy bits disproportionately damages performance both in and out of distribution. Finally, showing that even within the training distribution the removal of some bits can have a positive impact.

As models have increased in performance within the bounds of the i.i.d. assumption, recent years have seen growing interest in the OOD behaviour of machine learning systems. While many approaches have studied the effects of external changes to a model's training regime on OOD behaviour (e.g. domain randomization or auxiliary loss functions), to the best of our knowledge our proposal of the entropy of an intermediate representation as a guide to its effects OOD is a novel approach.

In Section 2 we outline the approach that we use to learn discrete representations of an input space using a contrastive unsupervised method. In Section 3 we present our methodology for removing parts of the model's intermediate representations with different 'masking strategies', and 4 we discuss the results from training and analyse the effects of distributional shift (Section 4.3). In Section 5 we review related work, and in Section 6 we conclude with a discussion of limitations and possible avenues for further work, for instance, applying the insights of our experiments to improve the out-of-distribution performance of downstream systems built on learned features.

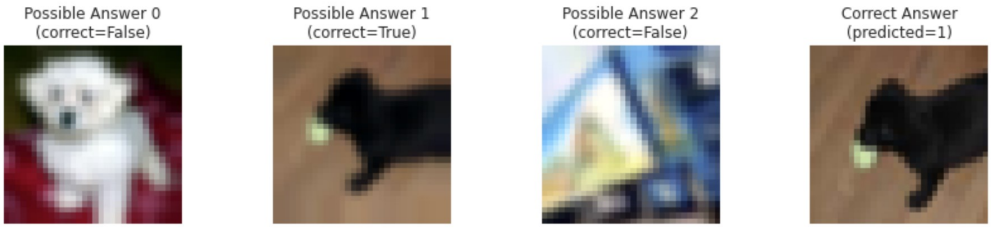


Figure 1: An example of a contrastive task ($k = 3$). For a given dataset, the distinguisher is shown k images, among which k-1 distractor images, and has to predict the correct image.

2 Contrastive Learning Methodology

2.1 Model Description

To learn representations of a domain we train an *encoder* network to produce a representation r of a given input x^*. This representation is given to a *distinguisher* network that is tasked with identifying x^* from a set of k images composed of x^* and k - 1 *distractor inputs* arranged randomly. We use the CIFAR-10 dataset (Krizhevsky, 2009) as the training distribution. The labels from the dataset are discarded and an unsupervised k-*contrast task* is constructed by pairing each image with k - 1 distractor images, shuffling, and giving the distinguisher k inputs to choose from. The same preprocessing is later used when out-of-distribution datasets are introduced. See Figure 1 for an example of a contrastive task and Figure 2 for the full architecture illustrated.

The encoder network is composed of a convolutional network (CNN) that takes a $32 \times 32 \times 3$ dimensional tensor as input (CNN_A in Figure 2), followed by: a 3×3 convolutional layer with 64 filters and ReLU activation; two 3×3 convolutional layers with 64 filters, ReLU activation, and a stride-length of 2; a flatten layer; and finally a dense layer without any activation that projects into $\mathbb{R}^{|r|}$, where $|r|$ is a hyperparameter controlling the 'representation length' of r. Next, between the encoder and the distinguisher, there is a *discretize/regularize unit* (Foerster et al., 2016). Following the literature in which this component was developed, we will refer to this as a *communication channel* (see in green in Figure 2). The channel is a differentiable unit that, during training, 'soft discretizes' activations passed through it by applying Gaussian white noise (GWN) and a sigmoid function. Then at test time we 'hard discretize' the activations by passing through a sigmoid function and emitting 0 if the result is less than 0.5 and 1 otherwise. This enables the end-to-end learning of a discrete representation via backpropagation from the output of the distinguisher. We configure the channel with a fixed GWN standard deviation of 0.5 during training.

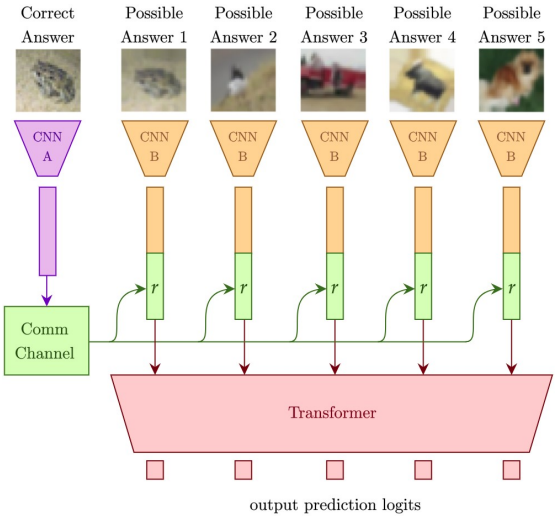


Figure 2: Architecture diagram ($k = 5$). The encoder is shown as the purple and green components, and the distinguisher is the orange and red.

The distinguisher network is composed of another convolutional network (CNN_B in Figure 1) with exactly the same input and layers as the CNN in the encoder (initialized separately and no parameter sharing), except projecting to a fixed embedding size of 128. This CNN is shared for each of the 'possible answer' images, producing embeddings that are each concatenated with the representation r from the encoder (i.e. the output of the communication channel) and fed into a transformer network (Vaswani et al., 2017) as tokens. The transformer is composed of two self-attention encoder layers with 3 heads of dimension 64, and a dropout rate of 0.1. After the transformer layers each token

Training k	Representation Length 64	128	256	512
3	0.909 ± 0.029	0.869 ± 0.015	0.870 ± 0.052	0.887 ± 0.015
5	0.797 ± 0.026	0.688 ± 0.077	0.759 ± 0.131	0.820 ± 0.166
10	0.866 ± 0.103	0.579 ± 0.018	0.643 ± 0.231	0.736 ± 0.171
20	0.662 ± 0.170	0.538 ± 0.230	0.532 ± 0.380	0.481 ± 0.337

Table 1: Accuracy on CIFAR-10 test set of trained models with different k and $|r|$ values.

is projected onto a single dimension without activation. This is then taken as the log-probability (logit) that the corresponding possible answer is correct. The networks are trained together with a sparse categorical crossentropy loss on these logits and the index of the correct answer. The use of a transformer and a shared encoder for the input images means that a model trained, for example, on a 3-contrast dataset ($k = 3$) can be tested on a 5-contrast dataset without any modification.

2.2 In-Distribution Training

We trained 54 independent encoder-distinguisher pairs[1] for 10 epochs on CIFAR-10 and removed models that did not converge, resulting in 51 trained models. Models were trained with varying combinations of representation lengths and number of distractors: $(|r|, k) \in \{64, 128, 256, 512\} \times \{3, 5, 10, 20\}$. See Table 1 for the test accuracy statistics for the models on the k-contrast CIFAR-10 training distributions. See the Supplementary Material for a full description of the training methodology.

2.3 Motivation for Discrete Representations

While the use of a communication channel to discretize the representations poses optimization challenges, it also provides a large benefit when it comes to computing the entropy values of each bit in the representation. The computation is reduced from approximating a continuous integral over the unit interval to a simple formula for the entropy of a binary variable, as outlined in Section 3.1. This allows us to run a greater number of experiments with higher precision than if we had used continuous representations.

2.4 Motivation for Task Choice

The choice of this unsupervised contrastive learning task as the setting for analysing our hypotheses comes down to two reasons. The first is that we need a task that can be easily transferred to different data distributions. A task such as image classification limits the available datasets as it requires the out-of-distribution testing data to have the same (or at least overlapping) image labels. On the other hand, a k-contrast task can be constructed from any unlabelled set of images. Another viable candidate task with the same property is autoencoding. However, autoencoding is a much harder task as intermediate representation in the bottleneck needs to incorporate more information. But more importantly, the contrastive task provides a finer degree of control on the entropy distributions within the intermediate representation r, as shown in the following sections. This control allows us to further investigate the implications of our proposals than would otherwise be possible with autoencoders.

3 Analysis Methodology

3.1 Entropy of Representation Bits

Each representation r produced by an encoder network consists of a number of bits $|r|$, referred to as the representation length. By considering each bit at index i as a random variable B_i we can compute the binary entropy of the bit on a given dataset $\mathcal{D}$:

$$H(B_i \mid \mathcal{D}) = -p \log_2 p - (1-p) \log_2 (1-p), \quad \text{where } p = P(B_i = 1 \mid \mathcal{D}). \tag{1}$$

[1] A sweep of 3 runs for each pair of $(|r|, k)$ plus 6 initial separate runs.

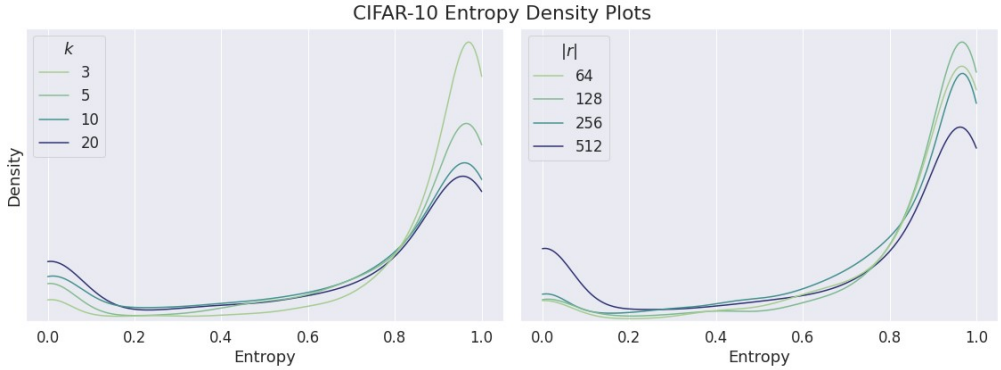


Figure 3: Bit-entropy distributions for models training with different representation length and training k parameters on the training distribution, plotted as kernel density estimate (KDE) plots[2]in Seaborn (Waskom, 2021). On the left, the distribution plots are stratified by k, and on the right by $|r|$.

Entropy close to 1 means that the bit is 0 or 1 with roughly equal probability of $p = 0.5$. Very low entropy means that the bit is either almost always 0 or almost always 1. In Figure 3 we see how these entropy values are distributed for the trained models with different parameters, when tested on the training distribution (CIFAR-10). We notice that for smaller representation lengths and/or few distractors the distribution tends to skew towards higher entropy bits. In separate experiments where we further varied representation lengths, we find that for smaller $|r|$ equal to 8, 16 or 32, all bits have entropy higher than 0.8, which makes studying bits based on entropy variation uninteresting for these representation lengths. For a visualization of these entropy values see the Supplementary Material. Representation lengths of 64, 128, 256 and 512 all lead to a wide range of entropy values.

A theoretical analysis of the optimal bit-entropy can be found in the Supplementary Material. We analyse the case that: the encoder and distinguisher need to decide on a communication protocol before interacting; each bit corresponds to one 'feature'; the encoder sends a 1 if a given feature is present and a 0 otherwise. In this case, a protocol where all bits have maximal entropy and are statistically independent is the optimal solution. However, empirically we find wider distributions of entropy values, including very low-entropy bits (even when discarding zero-entropy bits), echoing results from other work (Kharitonov et al., 2020).

We suspect that two important drivers towards the occurrence of low-entropy bits in Figure 3 are: 1) a larger encoding size allows for more redundancy and thereby the system can afford to ignore some bits; and 2) a larger set of distractors makes it more valuable to communicate specific features (on top of high entropy features) because the distinguisher needs to more finely differentiate between images.

3.2 Bit Masking Strategies

In this paper we are interested in the effects of strategically 'removing' parts of the model's intermediate representation, i.e. obscuring bits in r. This is achieved by means of a *masking variable* $m_i \in \{0, 1\}$ for each bit r_i in the representation. The masked bit $\hat{r}_i$ is then computed:

$$\hat{r}_i = m_i r_i + (1 - m_i)\frac{1}{2}. \tag{2}$$

In other words, when the masking variable $m_i = 0$ then $\hat{r}_i = 0.5$, and otherwise $\hat{r}_i = r_i$. In this paper we use three *masking strategies*; Random Masking, Top-Entropy Masking, and Bottom-Entropy Masking. In order to construct a mask with any of these strategies, we define a *masking proportion* p_{mask} that represents the percentage of bits in r that should be masked.

Firstly, to construct any mask $M = \{m_1, \ldots, m_{|r|}\}$ we will need to choose $l_{\text{mask}} = \lfloor\, p_{\text{mask}} \cdot |r| \,\rfloor$ bits to remove. For a *random mask* we draw l_{mask} masking variables from M at random with uniform

[2]While KDE plots are useful for highlighting the differences between distributions, which is the key reason we are using them here, they can also introduce unintended artefacts with bounded data. In this case, the downwards turn of the distributions on the right-hand side is not present in visualizations such as bar charts that more closely represent the raw data.

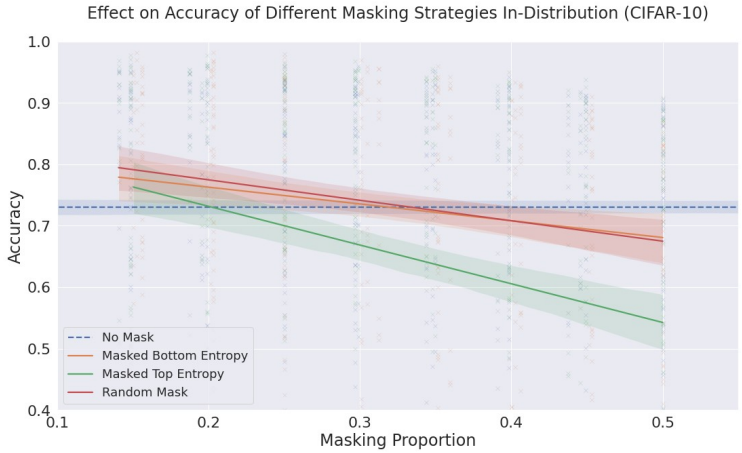


Figure 4: Accuracy for different masking strategies with varying masking proportions. For each strategy the raw values are plotted with crosses, but at a low transparency to avoid clutter and misleadingly giving the impression that whichever set that was plotted on top was dominating a region when it was not. Instead, to show the over all trends, linear regressions are rendered over the scatter. Finally, the dashed blue line represents no masking, which does not depend on any given masking proportion and therefore is only shown to provide a visual guide.

probability and without replacement, and set them to 0, we set the remaining $|r| - l_{\text{mask}}$ variables to 1. To construct a *top-entropy mask* we compute the entropy for each bit $h_i = H(B_i \mid \mathcal{D})$ and sort these values in descending order. We then take the bits associated with the first l_{mask} entropy values (i.e. highest entropy) and set their corresponding masking variables to zero. Likewise, for the *bottom-entropy mask* we take the last l_{mask} bits and remove those instead.

We place a constructed masking strategy between a given encoder and distinguisher and create an equivalent *masked model* by replacing each r_i with $\hat{r}_i$, before concatenating the representation to the output of the convolutional embeddings (Figure 2). We can then measure the mean accuracy of the masked model (which we will refer to as *masking accuracy*) for comparison to the unmasked model.

4 Experimental Results

In Section 4.1, we first discuss the effects of different masking strategies on CIFAR-10. In Section 4.2, analyze the changes to accuracy and to the entropy distributions when we apply our trained models to the OOD datasets. After which we analyse the effects of masking strategies when used OOD in Section 4.3, and present the result of masking low-entropy bits leading to improved OOD performance.

4.1 Analysis of Masking Effects In-Distribution (IND)

Before moving onto the out-of-distribution case, we will first examine the effects of applying the different masking strategies to the models that we trained on CIFAR-10, with the CIFAR-10 test data. For each of the 51 successfully trained models we evaluated the accuracy without any masking, and with each of the different masking strategies for masking proportions between 0.15 and 0.5 at 0.05 intervals. The results can be seen in Figure 4 for all of the data, and in Figure 5 which shows plots of the data separated by different values of k.

In this section we discuss three phenomena:

1. Removing top-entropy bits is more damaging than other masking strategies;
2. For low k all masking strategies have similar effects, whereas for large k the difference between the effects is quite pronounced; and
3. Especially for low k, masking out bits can *increase* accuracy.

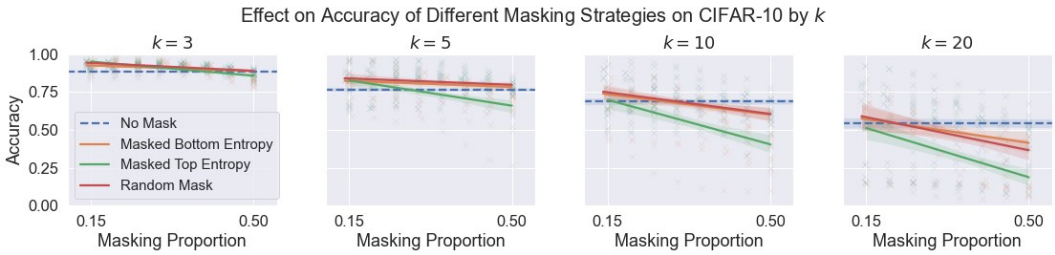


Figure 5: Accuracy for different masking strategies with varying masking proportions. One plot per value of k.

In Figure 4 we see that for any masking proportion, removing the top-entropy bits is more damaging to accuracy than masking out bottom-entropy bits. In light of general insights from information theory, this result is not too surprising. The highest entropy bits necessarily convey the most information, and so it follows that their removal should lead to the largest drop in performance.

Figure 5 shows that for low values of k the effects of all masking strategies are fairly similar, whereas for high k the effects come apart. One explanation for this phenomenon might be that, as observed in Figure 3, the lower-k models tend to have a higher density of high-entropy bits. Therefore even when we select bits from the lower end of the entropy distribution we are generally taking away more information from the distinguisher than we do for higher-k values. Thus masking from the top or the bottom removes bits with similar entropy values.

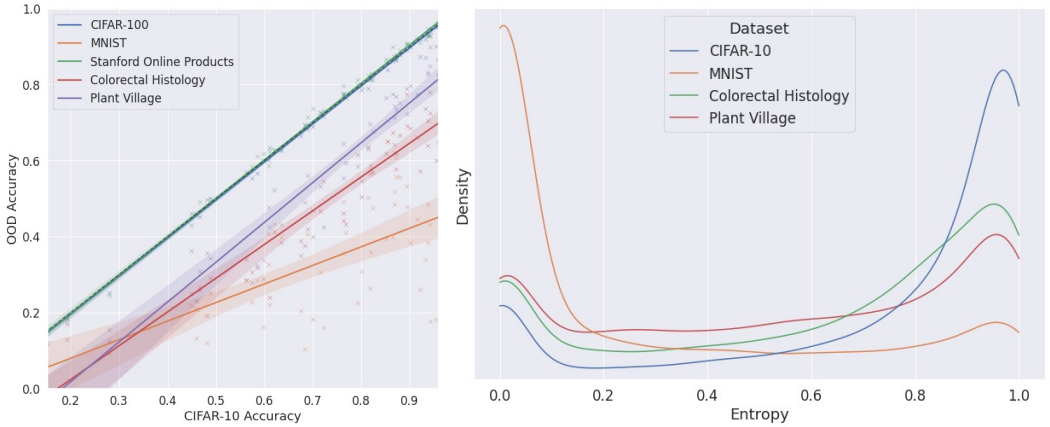
In general, we did not expect any of the masking strategies to provide a benefit when applied within the training distribution. Yet, in an unexpected turn of events we see that with a small enough masking proportion (around $p_{\text{mask}} < 0.3$) we see an *increase* in accuracy, especially with the random and bottom-entropy strategies (which yield roughly the same results).

In Figure 5 we break down how the masking accuracy improves for small p_{mask} in more detail. We see that the effect is in fact more pronounced for the lower-k values. This is especially remarkable because the higher starting accuracy values (i.e. with no mask) of the models trained with lower-k values suggests that there are only diminishing returns to be made in these cases. Our initial hypothesis was that the masking may be 'undoing' overfitting to the training set. But for each of the trained models we have verified that there is no overfitting (see the Supplementary Material for a visualization). One driver behind the relationship between k and masking accuracy could be that the contrastive task is easier for low k, in the sense that representations that convey less information about the target image should still allow the distinguisher to make the correct guess with high probability. This may cause more redundancy, which in turn may play a role in the increase in accuracy after masking.

4.2 Out-of-Distribution Behaviour

To evaluate the effects of distributional shifts we test our 51 trained models on the CIFAR-100 (Krizhevsky, 2009), Stanford Online Products (Song et al., 2016), Colorectal Histology (Kather et al., 2016), Plant Village (Hughes & Salathe, 2015), and MNIST (LeCun et al., 1999) datasets. These datasets were chosen to provide a range of different kinds of images, as judged subjectively. CIFAR-100, being drawn from the same subset of images as CIFAR-10, was chosen for its expected closeness to the training distribution. Stanford Online Products is a collection of photographs of items that contains a wide range of objects not encountered in CIFAR-10. Colorectal Histology is a medical dataset containing histological close-up photos of human tissue (un)affected with colorectal cancer. Plant Village contains photographs of individual healthy and unhealthy leaves presented on a neutral background. This dataset was chosen as an interesting challenge due to the potential familiarity that the model would have with leaves, as they appear in many CIFAR-10 images (e.g. in close-up photos of frogs on leaves). Yet, it would be unlikely for any of the models to have needed to distinguish between types of leaves during training. Finally, we have the MNIST dataset of digital handwritten numbers, which was chosen to test on non-photographic images.

In Figure 6 we show two visualizations that demonstrate the shift in behaviour that results from applying the models to the new datasets. In Figure 6a we plot the shift in accuracy values for each



(a) Accuracy Shift

(b) OOD Entropy Distributions

Figure 6: Two figures illustrating the effects of distributional shift on accuracy and entropy distributions.

dataset, following Taori et al. (2020). The $y = x$ line is plotted with a black dashed line, however, it is obscured by the regression lines for CIFAR-100 and Stanford Online Products. This tells us that there is no distributional shift for these datasets, i.e. no loss in performance. For this reason, we drop these datasets from all out-of-distribution analysis. For the other datasets, we see in order of increased degradation: Plant Village, Colorectal Histology, and MNIST.

Figure 6b shows the density of entropy values. We find that the entropy distribution of bits evaluated on MNIST has much more mass in the low entropy region than CIFAR-10. Bits of models evaluated on Colorectal Histology and Plant Village both follow more evened-out distributions.

4.3 Analysis of Masking Effects Out-of-Distribution (OOD)

In order to understand the effects of masking on accuracy in the OOD setting we measure the *mean change in accuracy* of a masking strategy under various circumstances. We also report the standard deviations associated with these estimates. As in the case of in-distribution masking we evaluated the masking strategies for a sweep of masking proportions between 0.15 and 0.5 at 0.05 intervals. We cut-off the maximum masking proportion $p_{\text{mask}} \leq 0.25$ for all further analysis as beyond that threshold masking has an almost universally negative effect. The overall mean accuracy changes can be seen in Table 2. We see that masking the bottom-entropy or random bits produces the highest increase, albeit with a large variance.

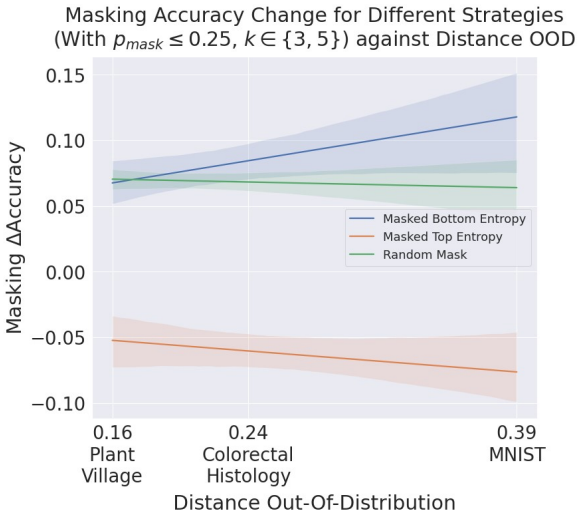


Figure 7: Effect of Masking Further OOD

This variance can be understood and disentangled by separating the low-k models from the high-k models. What we see is that the benefits of bottom-entropy masking are more prevalent for low-k models. This is visualized in Figure 8 where we illustrate the *effective robustness* of each of the masking strategies on the three OOD datasets. In the Supplementary Material we include plots for all values of k and p_{mask} that we tested. Effective robustness is a concept introduced by Taori et al. (2020) as a way to understand the efficacy of a method for

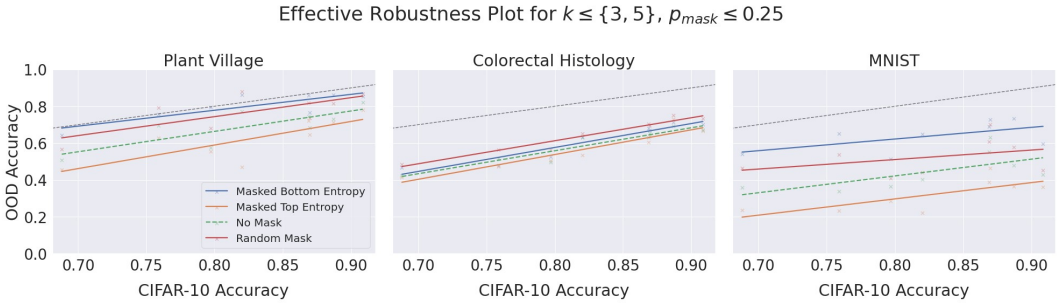


Figure 8: Effective robustness plots for low-k models. $y = x$ shown as black dashed line.

	CIFAR-10	Colorectal Histology	MNIST	Plant Village
Masked Bottom Entropy	1.6 ± 8.0	-2.0 ± 14.3	9.4 ± 15.6	3.0 ± 23.7
Masked Top Entropy	-4.3 ± 21.4	-7.8 ± 19.0	-16.6 ± 5.3	-18.5 ± 21.8
Random Mask	2.5 ± 12.3	3.4 ± 10.9	4.2 ± 13.7	2.1 ± 19.6

Table 2: Mean accuracy shift (in percentage points) after masking with each strategy. After running paired t-tests we find that all of these accuracy shifts are statistically significant (with $p = 0.05$).

increasing robustness to distributional shift. By plotting the baseline regression line for unaltered models with differing in-distribution accuracy values on the diagram we can observe whether a proposed robustness method moves towards the $y = x$ line (i.e. no degradation). Crucially, with these plots, we are able to account for each model's performance on the training distribution. Hence, despite the large variance in the performance of models trained across various k and $|r|$ values[3], we are able to discern the effects of the masking interventions.

In our case, we see that – as is consistent with previous results – for each dataset the top-entropy masking moves below the dashed green line showing the baseline unmasked models. On the other hand, the random masking and bottom-entropy masking lines move closer to $y = x$ (as compared to the no masking lines). For Plant Village we see that almost all of the in-distribution accuracy is recovered. For MNIST we find the most substantial jump, and the largest benefit of bottom-entropy over random masking.

The effect is further illustrated in Figure 7 in which we show the relationship between change in accuracy due to masking and what we have called 'distance out-of-distribution', i.e. the mean difference between CIFAR-10 accuracy and accuracy on the given dataset. In this plot we see the regression lines (with 95% confidence intervals) that show the trend in light of the magnitude of distributional shift away from CIFAR-10 for each of the test datasets. With this context, we most starkly see the effect that bottom-entropy masking can have for large distributional shifts.

5 Related Work

Our work adds to the toolkit of methods to aid in understanding and improving robustness to distributional shift, which for example includes forms of data augmentation (Hendrycks et al., 2021) and abstaining from making a prediction in the face of uncertainty (Thulasidasan et al., 2021). For a general overview of problems and methods in OOD robustness see Shen et al. (2015).

Below we reference some notable entropy-based methods that have a different purpose than improving OOD robustness. Chatterjee & Mishchenko (2019) low entropy (or "rare") signals to analyze extent to which a model is overfitted to the training distribution. Entropy-based methods have also been used widely in the adjacent problem of OOD detection. For example, predictive entropy measures the uncertainty of the prediction of a sample given a training distribution and is used to calculate the

[3]Accuracy ranging between 0.65 and 0.95 for even the high-performing low-k models, as shown in the x-axes of Figure 8

extent to which a sample is OOD (Kirsch & Gal, 2021). However, we apply entropy in an *entirely different context*, namely, we calculate the entropy of *latent variables* to estimate how robust they will be to distributional shift. Relative entropy (KL-divergence) is a popular measure and is notably used in the Bits-Back method (Hinton & van Camp, 1993; Flamich et al., 2020) to calculate the optimal compression rate in latent variables. Images that are traditionally compressed by a variational auto-encoder have now been compressed with code-length close to this theoretical optimum Flamich et al. (2020).

There are no clear benchmarks to compare our work with, as to our knowledge OOD robustness on an unsupervised contrastive task has not been studied before. Our motivation for using this task can be found in Section 2.4. There has been work applying unsupervised contrastive learning to improving few-shot classification generalization (Yang et al., 2022), and into self-supervised contrastive learning (Huang et al., 2021).

Contrastive representation learning takes many forms; in computer vision alone there are many approaches for applying deep learning to multiple inputs and producing representations to distinguish between them; see Jaiswal et al. (2020) for a review. For our work where we are dealing with a discrete channel between the encoder and distinguisher, the most closely related work is in the field of 'emergent communication' where deep learning systems are tasked with solving communication tasks (Lazaridou et al., 2017; Foerster et al., 2016). In this context, our work is viewed as a Lewis Signalling Game (Lewis, 1969) with the most similar set-ups being Lazaridou et al. (2017) and Kharitonov et al. (2020). The entropy of messages in learned communication has been studied by Kharitonov et al. (2020); Chaabouni et al. (2019), but not with a focus on distributional shift. Finally, in reinforcement learning (RL) Eysenbach et al. (2021) find that compressing observations for a RL agent improves the policy's robustness. Rather than masking out bits, they reduce the number of bits by adding a cost to communicating a bit.

6 Conclusion

In this paper we have investigated the out-of-distribution effects of using different strategies to remove bits from discrete intermediate representations in an unsupervised contrastive learning task. We have studied how the difficulty of the task impacts the entropy distribution of the learned representations and shown the following key results: 1) even in-distribution, removing parts of the intermediate representations can have a positive effect (Section 4.1); and 2) removing low-entropy bits can greatly improve the performance of models out-of-distribution (Section 4.3), notably almost entirely restoring in-distribution performance for one of our datasets (see Figure 8). Additionally, in line with theoretical results, we did find that across the board, removing high-entropy bits is more harmful than randomly removing bits or specifically removing low-entropy bits.

However, the results also present mysteries that prompt further experiments and analysis. At the time of writing, we do not have a clear understanding of why the removal of bits within the training distribution should increase performance, as we would expect the encoder to learn an optimal protocol. We have presented some hypotheses in Section 4.1 that suggest that a better understanding of how redundancy is encoded in the intermediate representations would be a helpful line of further inquiry. Relatedly, in light of viewing our task from the perspective of emergent communication, studying the relationship of our findings to *compositionality* within the representations, i.e. the interdependence and polysemanticity of bits, would provide valuable insight.

Next, there is a need for a deeper understanding of the conditions in which our results hold. Within our experimentation, we found that the effect (of harm from low-entropy features OOD) was less pronounced for models trained on the more difficult tasks (higher numbers of distractors). From our data, it is unclear if this relationship represents something fundamental or if it is a side-effect of these models generally performing to a lower standard. One of the most important avenues of further work is in testing if other systems built on top of the learned representations in this paper inherit the same OOD robustness under low-entropy masking.

To our knowledge, there are no existing state-of-the-art (SOTA) methods for OOD robustness in contrastive learning to benchmark our proposals against. When future studies investigate the effects of entropy-based masking in domains such as classification or control it will be important to compare our methods to OOD robustness benchmarks for those problems.

REFERENCES

Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Jia Yangqing, Rafal Jozefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dandelion Mané, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems, 2015. URL `https://www.tensorflow.org/`.

Rahma Chaabouni, Eugene Kharitonov, Emmanuel Dupoux, and Marco Baroni. Anti-efficient encoding in emergent communication. In H Wallach, H Larochelle, A Beygelzimer, F d Alché-Buc, E Fox, and R Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.

Satrajit Chatterjee and Alan Mishchenko. Coherent gradients: An approach to understanding generalization in gradient descent-based optimization, 2019.

Ben Eysenbach, Ruslan Salakhutdinov, and Sergey Levine. Robust predictable control. In Marc'Aurelio Ranzato, Alina Beygelzimer, Yann N. Dauphin, Percy Liang, and Jennifer Wortman Vaughan (eds.), *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pp. 27813–27825, 2021.

Gergely Flamich, Marton Havasi, and José Miguel Hernández-Lobato. Compressing images by encoding their latent representations with relative entropy coding. *CoRR*, 2020. URL `https://arxiv.org/abs/2010.01185`.

Jakob Foerster, Ioannis Alexandros Assael, Nando de Freitas, and Shimon Whiteson. Learning to Communicate with Deep Multi-Agent Reinforcement Learning. In D. D. Lee and M. Sugiyama and U. V. Luxburg and I. Guyon and R. Garnett (ed.), *Advances in Neural Information Processing Systems 29*, pp. 2137–2145. Curran Associates, Inc., 2016.

Dan Hendrycks, Steven Basart, Norman Mu, Saurav Kadavath, Frank Wang, Evan Dorundo, Rahul Desai, Tyler Zhu, Samyak Parajuli, Mike Guo, Dawn Song, Jacob Steinhardt, and Justin Gilmer. The many faces of robustness: A critical analysis of out-of-distribution generalization. In *2021 IEEE/CVF International Conference on Computer Vision, ICCV 2021, Montreal, QC, Canada, October 10-17, 2021*, pp. 8320–8329. IEEE, 2021.

Geoffrey E. Hinton and Drew van Camp. Keeping the neural networks simple by minimizing the description length of the weights. New York, NY, USA, 1993. Association for Computing Machinery. ISBN 0897916115.

Weiran Huang, Mingyang Yi, and Xuyang Zhao. Towards the generalization of contrastive self-supervised learning. *CoRR*, abs/2111.00743, 2021. URL `https://arxiv.org/abs/2111.00743`.

David P Hughes and Marcel Salathe. An open access repository of images on plant health to enable the development of mobile disease diagnostics through machine learning and crowdsourcing. *CoRR*, abs/1511.08060, 2015.

Ashish Jaiswal, Ashwin Ramesh Babu, Mohammad Zaki Zadeh, Debapriya Banerjee, and Fillia Makedon. A Survey on Contrastive Self-supervised Learning. *Technologies*, 9(1):2, 10 2020. doi: 10.48550/arxiv.2011.00362.

Jakob Nikolas Kather, Cleo-Aron Weis, Francesco Bianconi, Susanne M Melchers, Lothar R Schad, Timo Gaiser, Alexander Marx, and Frank Gerrit Zollner. Multi-class texture analysis in colorectal cancer histology. *Scientific Reports (Nature Publishing Group)*, 6:27988, 2016.

Eugene Kharitonov, Rahma Chaabouni, Diane Bouchacourt, and Marco Baroni. Entropy Minimization In Emergent Languages. In *Proceedings of the 37th International Conference on Machine Learning, PMLR 119*, pp. 5220–5230, 2020.

D.P. Kingma and L.J. Ba. Adam: A Method for Stochastic Optimization. In *International Conference on Learning Representations (ICLR)*, San Diego, CA, USA, 2015.

Mukhoti Jishnu Amersfoort Joost Torr Philip H.S. Kirsch, Andreas and Yarin Gal. On pitfalls in ood detection: Entropy considered harmful. *In Uncertainty Robustness in Deep Learning at Int. Conf. on Machine Learning (ICML Workshop)*, 2021.

Alex Krizhevsky. Learning multiple layers of features from tiny images. Technical report, 2009.

Angeliki Lazaridou, Alexander Peysakhovich, and Marco Baroni. Multi-Agent Cooperation and the Emergence of (Natural) Language. In *The International Conference on Learning Representations (ICLR)*, 2017.

Yann LeCun, Corinna Cortes, and Chris Burges. MNIST handwritten digit database, 1999. URL `http://yann.lecun.com/exdb/mnist/`.

David K. Lewis. *Convention: A Philosophical Study*. Wiley-Blackwell, Cambridge, USA, 1969. doi: 10.2307/2218418.

Zheyan Shen, Jiashuo Liu, Yue He, Xingxuan Zhang, Renzhe Xu, Han Yu, and Peng Cui. Towards Out-Of-Distribution Generalization: A Survey. *Journal of Latex Class Files*, 14(8), 2015.

Hyun Oh Song, Yu Xiang, Stefanie Jegelka, and Silvio Savarese. Deep Metric Learning via Lifted Structured Feature Embedding. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

Rohan Taori, Achal Dave, Vaishaal Shankar, Nicholas Carlini, Benjamin Recht, and Ludwig Schmidt. Measuring Robustness to Natural Distribution Shifts in Image Classification. In *The 34th Conference on Neural Information Processing Systems (NeurIPS 2020)*, Vancouver, Canada, 2020.

TF Devs. TensorFlow Datasets: A collection of ready-to-use datasets, 2022. URL `https://www.tensorflow.org/datasets`.

Sunil Thulasidasan, Sushil Thapa, Sayera Dhaubhadel, Gopinath Chennupati, Tanmoy Bhattacharya, and Jeff A. Bilmes. An effective baseline for robustness to distributional shift. In M. Arif Wani, Ishwar K. Sethi, Weisong Shi, Guangzhi Qu, Daniela Stan Raicu, and Ruoming Jin (eds.), *20th IEEE International Conference on Machine Learning and Applications, ICMLA 2021, Pasadena, CA, USA, December 13-16, 2021*, pp. 278–285. IEEE, 2021.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention Is All You Need. In *The 31st Conference on Neural Information Processing Systems (NIPS 2017)*, Long Beach, CA, USA, 2017.

Michael L Waskom. seaborn: statistical data visualization. *Journal of Open Source Software*, 6(60): 3021, 2021. doi: 10.21105/joss.03021.

Jiawei Yang, Hanbo Chen, Jiangpeng Yan, Xiaoyu Chen, and Jianhua Yao. Towards better understanding and better generalization of low-shot classification in histology images with contrastive learning. In *International Conference on Learning Representations*, 2022.