

Table 2: Sample true feature firings and corresponding SAE latent activations. Feature 1 only fires if feature 0 fires. Feature 0 has variance of 0.1 in its firing magnitude, while the other feature have no variance in their firing magnitude.

TRUE FEATURES				SAE LATENT ACTS			
1.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00
1.00	1.00	0.00	0.00	1.27	0.00	0.00	0.22
0.90	1.00	0.00	0.00	1.27	0.00	0.00	0.12
0.75	1.00	0.00	0.00	1.26	0.00	0.00	0.00

We call this phenomenon **partial absorption**. In partial absorption, there’s co-occurrence between a dense and sparse feature, and the sparse feature absorbs the direction of the dense feature. However, the SAE latent tracking the dense feature still fires when both the dense and sparse feature are active, only very weakly. If the magnitude of the dense feature drops below some threshold, it stops firing entirely.

Feature absorption is an optimal strategy for minimizing the L1 loss and maximizing sparsity. However, when a SAE absorbs one latent into another, the absorbing latent loses the ability to modulate the magnitudes of the underlying features relative to each other. The SAE can address this by firing the latent tracking the dense feature as a "correction" to add back some of the dense feature direction into the reconstruction. Since the dense feature latent is firing weakly, it still has lower L1 loss than if the SAE fully separated out the features into their own latents.

Imperfect co-occurrence can still lead to partial absorption Next, we test what will happen if feature 1 is more likely to fire if feature 0 is active, but can still fire without feature 0. We set up feature 1 to fire with feature 0 95% of the time, but 5% of the time it can fire on its own. For this experiment, all features fire with magnitude 1.0 and 0 variance. We show the cosine similarities of the SAE encoder and decoder with true features in Figure 11. Some sample feature firings and corresponding SAE activations are shown in Table 3.

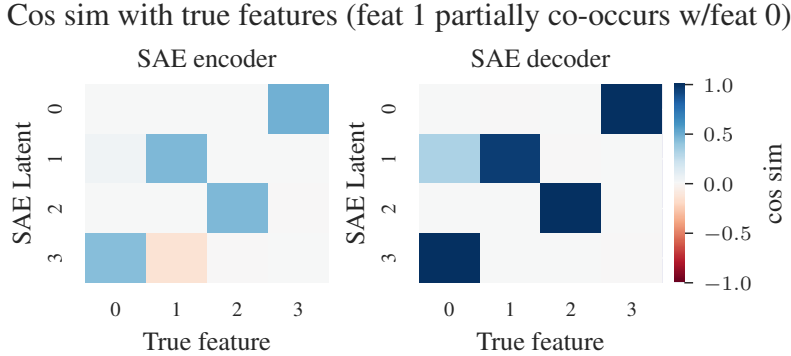


Figure 11: SAE encoder and decoder with true features. Feature 1 fires with feature 0 95% of the time, but 5% of the time feature 1 is allowed to fire on its own. Even though there is not perfect co-occurrence between features 0 and 1, we still see the SAE encoder and decoder learn a weak absorption pattern, with the decoder latent for feature 1 absorbing part of feature 0, and the encoder latent for feature 0 including a negative component of feature 1.

We see signs of partial absorption here as well. We see the same absorption pattern in the SAE encoder and decoder as we saw in our other absorption examples, although less severe than the previous examples. We also see in the sample firing patterns that when both feature 0 and 1 fire together, the latent tracking feature 0 fires with noticeably lower magnitude than when feature 0 fires on its own. Here, even though the co-occurrence between features 0 and 1 is not perfect, we still see partial absorption.

Absorption also affects TopK SAEs So far, we have only shown feature absorption occurring with standard L1 SAEs. Next, we examine how other absorption affects other architectures using

Table 3: Sample feature values and corresponding SAE activations. Feature 1 can only fire if feature 0 is active 95% of the time, but 5% of the time feature 1 can fire on its own. We see signs of partial absorption, where the latent tracking feature 0 fires noticeably more weakly if feature 1 is active.

TRUE FEATURES				SAE LATENT ACTS			
1.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00
1.00	1.00	0.00	0.00	0.00	1.05	0.00	0.67
0.00	1.00	0.00	0.00	0.00	0.95	0.00	0.00

a batch topk SAE [3]. Batch topk SAEs are an improved version of topk SAEs [10] where the top $k * B$ latents are used to reconstruct the SAE input, where B is the batch size. As the topk function enforces sparsity, there is no additional L1 loss term.

Topk SAEs are harder to use for very small toy models like our 4-feature toy model above, since if the k is too large relative to the size of the SAE the SAE will not learn correct features. To address this, we use a slightly larger toy model with 12 mutually orthogonal true features. All features fire independently with probability 0.15, except for the first 2 features. Feature 0 is the parent feature in our hierarchy, and fires with probability 0.4. Feature 1 is the child feature, and fires with probability 0.6 only if feature 1 fires, but never fires if feature 1 does not fire. All features fire with magnitude 1.0. We train a batch topk SAE with $k = 2$. We show the cosine similarities of the SAE encoder and decoder with true features in Figure 12.

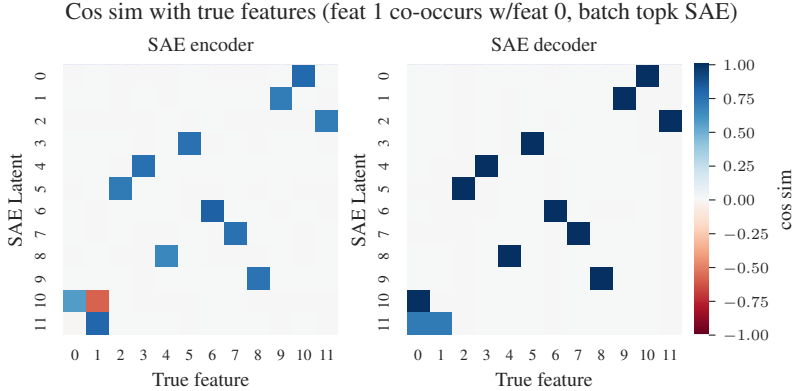


Figure 12: SAE encoder and decoder with true features for a batch topk SAE with $k=12$. Feature 1 is only allowed to fire if feature 0 fires. We still see a very clear absorption pattern between the latents tracking features 0 and 1 despite the lack of L1 loss.

We still see a clear absorption pattern between the latents tracking features 0 and 1 despite the lack of L1 loss. Absorption increases sparsity, which allows the topk SAE to have better reconstruction loss at a given k , and is thus what the SAE learns.

A.4 Ablation algorithm

A.5 How good is Gemma-2 on character identification tasks?

We evaluate how well can Gemma-2-2B identify the first letter or all the letters in a token (spelling the full token). We evaluate the accuracy of the model on all tokens in the LR probe validation set with a prompt containing 10 in-context examples selected at random from the full vocabulary. Our results are shown in Figure 13.

We see that performance on the first-letter identification task is high throughout token length, while the full-word spelling performance decreases as the length of the token increases.

Algorithm 1 SAE Latent Ablation

```
1: Insert SAE in model computation, including error term
2: Define a scalar metric on the model's output distribution
3: Calculate baseline metric value for a test prompt
4: for each token of interest do
5:   for each SAE latent do
6:     Set the SAE latent activation to 0
7:     Recalculate the metric
8:     Compute ablation effect (baseline - new metric)
9:     Reset the SAE latents to its original value
10:  end for
11: end for
```

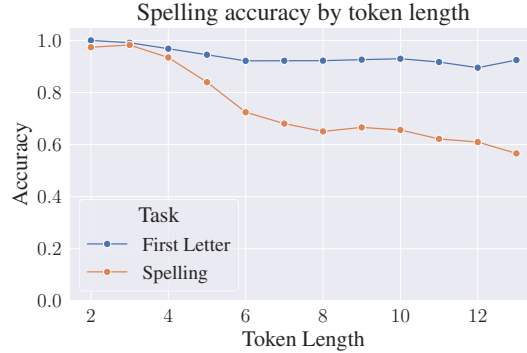


Figure 13: Baseline performance for Gemma-2-2B on first-letter identification and full-token spelling by token length.

A.6 Intervening on the first letter

If the model is using the identified SAE latents for predicting the first letter we should also be able to change what first letter it predicts just by changing the activations. For this experiment we use the SAE latents most cosine similar with the LR probe for the true first letter and for a new randomly selected letter. We take the intermediate activations of Gemma-2-2B in the residual stream and encode them using the SAE. Then we zero out the activation of the SAE latent associated with the original letter and change the activation of the SAE latent associated with the new letter into the average activation it has on tokens starting with this new letter.

Editing works better with latents from the narrower 16k SAE compared to the 65k, with the best L0s in the 75-150 range. This corresponds to the observed pattern of these SAE latents having higher F1 scores for classification. We report the results in Figure 14. The best SAEs on the layers 7-9 can achieve a substantial replacement, but note that the averages hide variance across individual tokens, where some get edited completely and others get unaffected. The edit success also varies based on the true first letter and the random new letter; for illustration we show a breakdown by letter for two specific SAEs in layer 7 in Figure 15.

A.7 Probe cosine similarity vs k=1 sparse probing

The first step when searching for a SAE latent that acts as a first-letter classifier involves searching for SAE latent which best acts as a classifier. In Figure 4, we achieve this by first training a LR probe on the first-letter task and using cosine similarity between that probe and the SAE encoder to find the best latent for the first-letter task. We also investigated using k-sparse probing with k=1 to select the best SAE latent instead. This involves training a linear probe with L1 loss and selecting the latent with the highest positive weight from the probe.

We find that both k=1 sparse probing yield nearly identical results, as seen in Figures 16 and 17. Additionally Figure 18 shows the cosine similarity of the LR probe with each SAE latent by letter for the canonical Gemma Scope layer 0 16k width SAE. In most cases there is an obvious probe-aligned