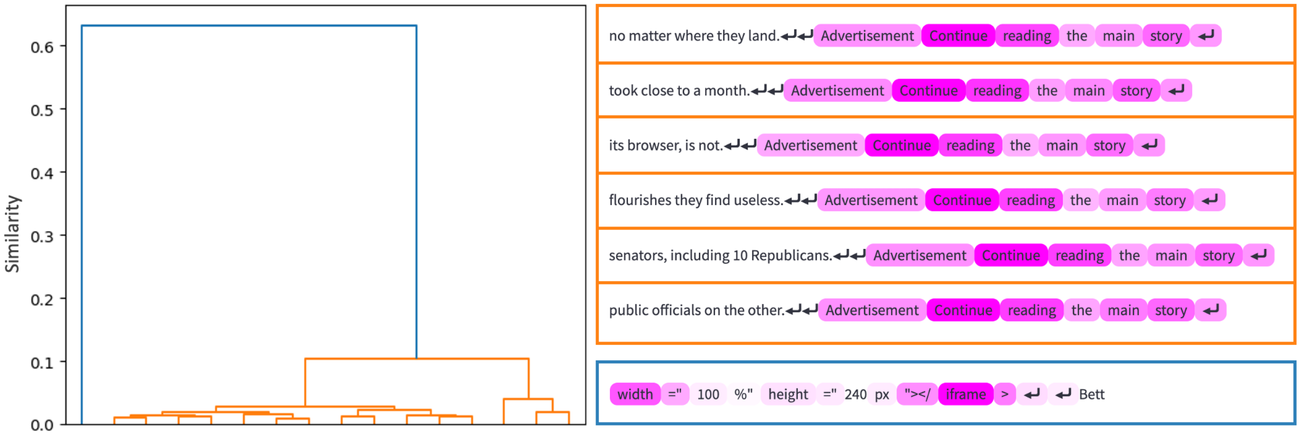

no matter where they land.↵↵ Advertisement Continue reading the main story ↵

took close to a month.↵↵ Advertisement Continue reading the main story ↵

its browser, is not.↵↵ Advertisement Continue reading the main story ↵

flourishes they find useless.↵↵ Advertisement Continue reading the main story ↵

senators, including 10 Republicans.↵↵ Advertisement Continue reading the main story ↵

public officials on the other.↵↵ Advertisement Continue reading the main story ↵

width =" 100 %" height =" 240 px "></ iframe > ↵ ↵ Bett

Figure 3. An example of a neuron with a common primary behaviour (orange) and a rare secondary behaviour (blue).

Table 2. Evaluation metrics for SAEs trained with and without the neuron embedding (NE) loss. Accuracy loss is the absolute drop in accuracy after ablating the MLP neurons with the reconstructed activations from the SAE, from a starting accuracy of 94.0%.

	MSE	L1	L0 / %	Acc. Loss / %
Standard	0.33	2600	2.4	1.5
+ NE Loss	0.55	3300	9.1	4.6

4.2. Training Sparse Auto-Encoders

We provide a proof-of-concept showing how we can integrate information from neuron embeddings into the loss function when training Sparse Auto-Encoders (SAEs). We first train an MLP with one hidden layer containing 64 neurons on the MNIST dataset (Deng, 2012) for 3 epochs, until the loss converges. We then experiment with training an SAE for the hidden layer with and without the neuron embedding loss term, and measure the effect on various evaluation metrics.

Each SAE has a hidden dimension of 512 (i.e., 8× the MLP hidden dimension), and we train them for one epoch over the training set. When incorporating the neuron embedding loss, we train for 200 steps ($\sim 40\%$ of an epoch) without the loss, then switch it on. This should allow the SAE neurons to stabilise, at which point the neuron embedding loss could be useful for pushing them to be more monosemantic.

We provide a UI [5] which allows a user to examine any neuron in the MLP hidden layer or the SAE, and provides visualisations for interpreting the neuron's behaviour, which can be used to understand how the SAE neurons differ between the two models.

Tables 2 and 3 show a suite of evaluation metrics for the SAEs trained with and without the neuron embedding (NE) loss. Table 2 shows the typical evaluation metrics that measure reconstruction error and activation sparsity, measured on the held-out test data. Adding the NE loss increases the reconstruction error, reflected in the increased mean-squared error on the reconstructed activations and the significantly greater drop in absolute accuracy when ablating the MLP activations with the SAE reconstructions. It also decreases the sparsity, with the percentage of active neurons per input almost quadrupling from 2.4% to 9.1% and the L1 loss increasing as well. Note that decreasing sparsity is not intrinsically good or bad, but in the normal regime it is typically associated with decreased interpretability.

Table 3. Additional evaluation metrics for SAEs trained with and without the neuron embedding (NE) loss. Distances are measured on the neuron embeddings of each neuron's test set dataset examples, and size is the number of test set examples that induce a non-zero activation for a neuron. Dead refers to the percentage of neurons which don't activate for any example from the training dataset.

	Max Dist	Mean Dist	Size	Dead / %
Standard	0.45	0.21	7	23.8
+ NE Loss	0.28	0.08	32	3.7

In contrast, Table 3 shows improvements in the distance metrics. For each SAE neuron, we collected the test examples which caused a non-zero activation (up to a maximum of 100 examples) and computed their neuron embeddings. We then measured the max and mean distance between points for each set of neuron embeddings, and took the median of these values over all neurons. We see significant decreases in the average max and mean distance between embeddings, which indicates potential improvements in monosemanticity.

Interestingly, this is in spite of the decrease in sparsity and a corresponding increasing in the average number of activat-

[5] https://mechmnistic.streamlit.app