Emergent organization of receptive fields in networks of excitatory and inhibitory neurons

SUPPLEMENT

A Tunings under the LIF model

The LIF model is based on the neuron dynamics of Keane and Gong (2015), and is the basis for the activation model. It interleaves excitatory and inhibitory neurons on a single grid, rather than having two "sheets" of neurons, one excitatory the other inhibitory (see Figure 1). The "activation" of a neuron in the LIF model is the number of times it fires over the simulation period, but the same update rule is used as for the activation model. When tuning receptive fields, the results were much more sensitive to the settings of the parameters under the LIF model than the activation model. Moreover, the trained edge detectors were blurry and not properly localized. We discuss how we addressed this problem later.

Computation under the dyanmics of Keane and Gong (2015) also made the LIF model far more computationally expensive. This was due to the small time steps and long simulation periods needed to simulate propagating waves. A small time step is desirable because of the nature of the discontinuous dynamical system in question, while a long simulation period is usually necessary to observe propagating wave patterns. In order to address these issues, we made several modifications to their model.

The first set of modifications we made were to shorten runtime. Keane and Gong (2015) simulate their model for 150,000 time steps (simulating 7.5 s in real time) while allowing for a transient phase of about 30,000 steps (1.5 s) on a grid of 300×300 neurons. For the sake of computation time, we decided to reduce this model to 30×30 neurons. In doing so, we found that we only needed to collect firing counts for about 4,000–10,000 time steps and allow for a transient phase of about 200 time steps until firing patterns stabilized. The modifications we discuss below allowed us to reduce this to just 50 time steps.

The LIF model is very sensitive to scalings of the terms in equations (1) and (3). Of particular importance is the scaling of the terms in the stimulus. The stimulus is based on the inner product between the image vector $X(t)$ and the feedforward weights Φ_{ij}. If this term is negative, we set it to zero. Through a grid search, we found it was optimal to scale this quantity by 10^{-4} before adding a fixed amount $F^E = 15\mu S$ in order to achieve a stable decrease in the model's reconstruction error and learn edge detectors arranged in a topographic map. Thus, the minimum external excitatory stimulus a neuron receives is F^E, and this can increase if the image is similar to its feedforward weight Φ_{ij}.

In order to improve the quality of the edge detectors, we made changes to the neuron dynamics. The first was to adapt the firing threshold of our set of neurons, inspired by the SAILnet model of Zylberberg et al. (2011). We calculated the firing rate, p_1, to be the average number of firings per neuron over the simulation period. To achieve convergence of the firing rate, we initially set V_T to a large value, like $V_T = 0\ mV$, and update it after each simulation period according to

$$V_T \leftarrow V_T - \delta(p^* - p_1),$$

where p^* is the target firing rate and δ controls the learning rate. If $p_1 > 1.1 \cdot p^*$, which may occur during the initial training steps, we skip the update to the receptive fields given by equation (9).

We were able to further increase sharpness of the edge detectors by modifying the postsynaptic conductance term $G(t)$. Keane and Gong (2015) define it as the difference of two exponential terms, varying based on whether the neuron is inhibitory or excitatory. This selection is very similar to the postsynaptic conductance of neurons, but introduces additional complexity through a long time-depedence on previous firings. We simplified the postsynaptic conductance to be $G(t) = \mathbb{1}(t = 0)$, thus making $G(t)$ depend only on a single time step. In line with this change, we shortened each neuron's refractory period from 100 time steps to just one. After making these modifications, we also observed that we could ignore the transient period advised by Keane and Gong (2015) and reduce the number of time steps during which we count firings to just 50.

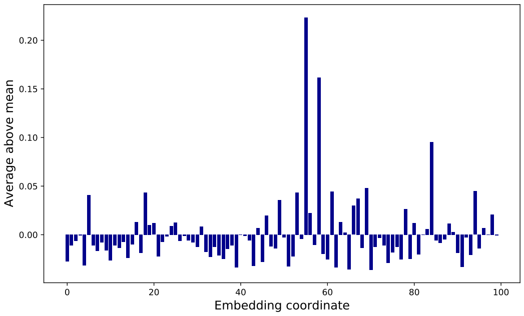

Fig. 1: Average absolute values of different entries in GloVe embedding.

The changes above enabled the LIF model to learn a topographic map with the desired pinwheel patterns, though it was still sensitive to parameter settings. After performing a grid search, we found that the most important conditions were that the target firing rate p^* be 3–5% and the excitatory be radius be 2 neurons. (We only explored 30×30 grids, these numbers will likely vary as the grid size increases.) Computing receptive fields for the LIF model differs slightly from the activation model. Because its activations cannot be negative, we cannot measure inhibition. To address this we compute two sets of activations: the first inputs delta functions having a spike in a given pixel position and zeros elsewhere, and standardizes the stimulus to have mean zero and variance one; the second inputs delta functions multiplied by -1, and is standardized the same way. We compute the activations for both of these inputs. The difference between the first and second activations gives the receptive fields of the neurons. Figure 3 shows an example of such a topographic map, along with the specific parameters used.

B Text experiment

B.1 Details of experiment with GloVe embeddings

To sample words, we use the Google 1-gram data from `http://storage.googleapis.com/books/ngrams/books/datasetsv2.html` and record the frequencies for words that have more than a million occurrences. We then include words that are in the GloVe vocabulary, and map each word w to its 100-dimensional embedding vector $\phi(w)$. This results in a vocabulary of 55,529 words. In the process of tuning the receptive fields, we sample words according to the unigram distribution. As shown in Figure 1, we find that average absolute values of components 55, 58, and 84 are significantly higher than the others; we thus remove these entries from the embeddings. In the training process, we sample the words according to the unigram distribution, resulting in a 97-dimensional input stimulus.

B.2 Details of experiment with embeddings derived from an LSTM language model

To sample words, we use the WikiText-103 data from `https://s3.amazonaws.com/research.metamind.io/wikitext/wikitext-103-v1.zip`, with a vocabulary of 267,744 words, and record each word's frequency. We then build an LSTM language model with this WikiText-103 data using the `fairseq` sequence modelling toolkit (Ott et al., 2019). We extract the first layer of the model, which results in a 300-dimensional word embedding as the input stimulus. Similar to the preceding experiment, to train our model, we sample words according to the unigram distribution.

The structure of the excitatory and inhibitory network of neurons and the hyperparameters of our model are the same as for the preceding experiment, and the tunings shown in Figure 2 organized in the grid of 40×40 excitatory neurons is qualitatively similar to that in Figure 3 in the main text.

Code

The code for all experiments is available in a zip file.

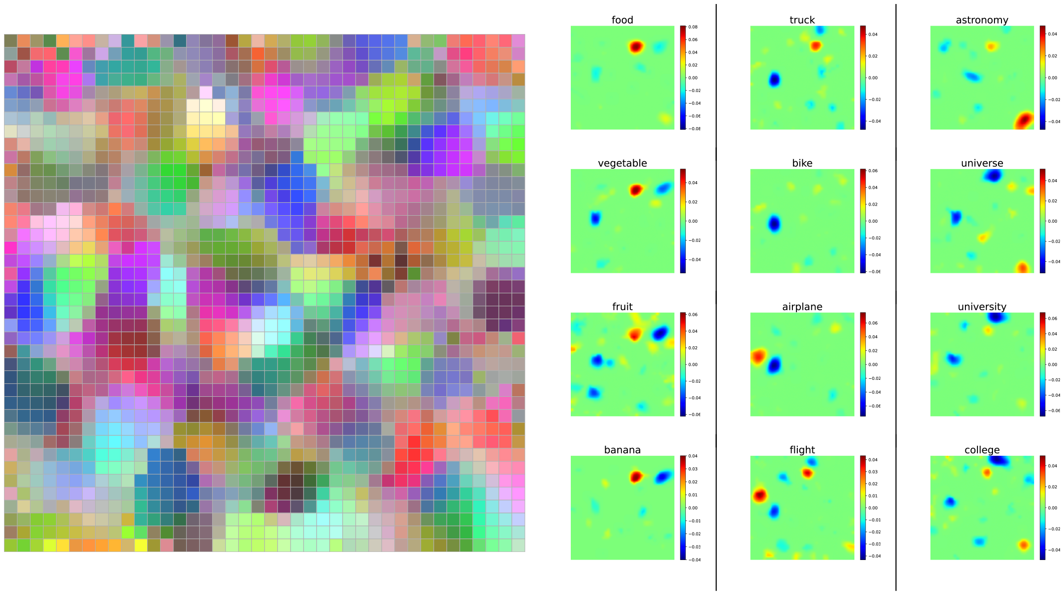

Fig. 2: Tunings of a grid of 40 × 40 excitatory neurons from stimuli that are embedding vectors of words, sampled from the unigram distribution on a large corpus of text (WikiText-103). Left: each neuron is mapped to a color by taking the top 3-principal components of the receptive fields as an RGB value. Right: Neural activations for selected groups of words.

References

Keane, A. and Gong, P. (2015). Propagating waves can explain irregular neural dynamics. *Journal of Neuroscience*, 35(4):1591–1605.

Ott, M., Edunov, S., Baevski, A., Fan, A., Gross, S., Ng, N., Grangier, D., and Auli, M. (2019). fairseq: A fast, extensible toolkit for sequence modeling.

Zylberberg, J., Murphy, J. T., and DeWeese, M. R. (2011). A sparse coding model with synaptically local plasticity and spiking neurons can account for the diverse shapes of V1 simple cell receptive fields. *PLOS Computational Biology*, 7:1–12.

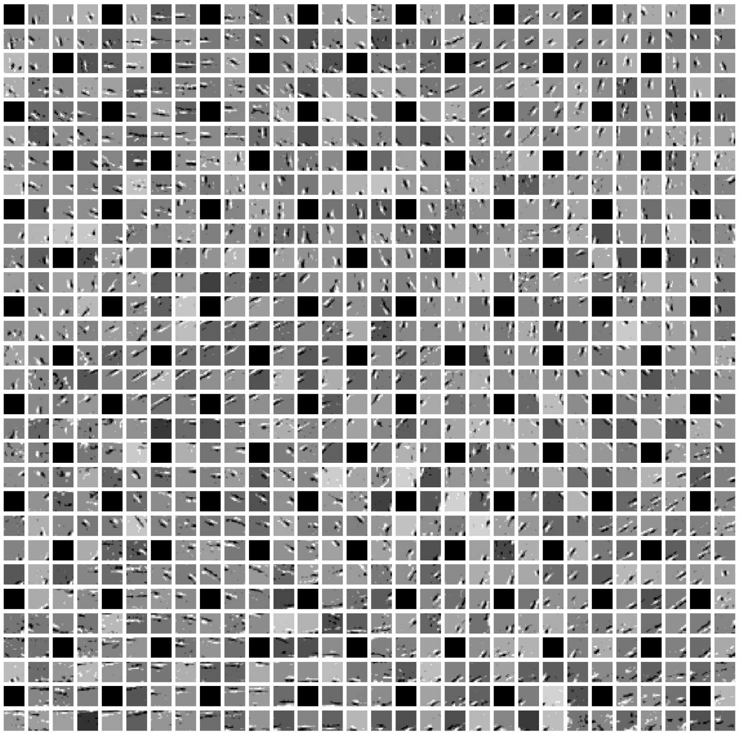

Fig. 3: Receptive fields for a 30×30 grid of neurons in the LIF model; black regions indicate inhibitory neurons. Parameters for training: image size, 16×16 pixels; batch size, 256; target firing rate, 3%; excitatory weight, 14.0; inhibitory weight, 10.0; excitatory radius, 2; inhibitory radius, 5; refractory period, 50 μs; gradient steps, 100,000.