Investigating Neuron Behaviour via Dataset Example Pruning and Local Search

Alex Foote

Interpretability Hackathon Report

Apart Research

PIs: Esben Kran, Neel Nanda, Fazl Barez

Date: 13th November, 2022

Abstract

This report presents methods for pruning and diversifying dataset examples that strongly activate neurons in a language model, to facilitate research into understanding the behaviour of these neurons. The pruning algorithm takes a dataset example that strongly activates a specific neuron and extracts the core sentence before iteratively removing words, to find the shortest substring that preserves a similar pattern and magnitude of neuron activation. This removes extraneous information, providing a much more concise input that is easier to reason about. The extracted substring, referred to as a Minimal Activating Example (MAE), is then used as a seed for local search in the input space. Using BERT, each word in the MAE is replaced by its most probable substitutes, and neuron activation is re-assessed. This creates positive and negative inputs that shed much more light on neuron behaviour than dataset examples alone. In two case studies we identify neuron behaviours that were not obvious from the raw dataset examples using this combination of pruning and local search. These methods could facilitate and significantly speed up research into neuron behaviour in language models, which is a key aspect of model interpretability.

Investigating Neuron Behaviour via Dataset Example Pruning and Local Search

Motivation

In this report we introduce a method for investigating which inputs are highly activating to a chosen neuron in a language model, to enable users to better investigate the role of these neurons. Understanding neurons at this level has been a crucial part of previous interpretability work, in both vision models [Olah et al.] and language models [Elhage et al.]. However, unlike in vision models, feature visualization via gradient descent is yet to be developed, due at least in part to the discrete nature of the input space of language models. Instead, dataset examples are often used [Nanda, Elhage et al.] - the model is run on a large amount of text and for each neuron the prompts that induced the greatest activation are stored. Exploring these examples can be informative for understanding the role of a neuron, but they generally contain extraneous information, slowing down the investigation. Additionally, they often lack diversity, and do a poor job of illustrating the full set of inputs that activate a neuron, making it difficult to pinpoint the exact aspects of the input to which the neuron is responding. As interpreting neurons is a key step in interpreting language models, and there are a very large number of neurons in modern language models (millions for the smallest to hundreds of billions for the largest), improvements to this process could have a significant impact by increasing the rate at which researchers can formulate and verify hypotheses about neuron function.

Methods

Pruning

To address the first issue of extraneous information in dataset examples, we developed a simple method for pruning a full dataset example to a Minimal Activating Example (MAE) - the shortest substring of the example which induces a similar pattern and magnitude of activation in the chosen neuron as the full example. To do this, we first split the example into sentences, and extract the sentence that contains the token with the highest activation, which we will refer to as the anchor token. We run the model on the extracted sentence and measure the activation on the anchor token to check that it still induces an activation that is at most *x%* less than the original activation, where *x* is a parameter that we set to 60%, based on initial experiments. We then iteratively remove a token from the end of the sentence, and again perform the activation drop-off test. We take the shortest substring (by number of tokens) that passes this test. We then repeat this process, now pruning tokens from the start of the shortest substring, and output the final MAE - the shortest substring that passes the activation test. Note that we also provide a window parameter, which will preserve a window of *w* tokens either side of the anchor token. This can help provide a small amount of guaranteed context, which can make the MAE more readable, and provide more context for the next stage of the process, the input space search.

We believe that MAEs are easier and faster to work with, as they are generally significantly shorter than the full dataset examples, and only provide the relevant context that has a direct impact on the activation of the neuron. For example, the method prunes the 0th example of neuron 2 in layer 3 of solu-8L-old to **“am not a native English speaker”**, a significantly more concise input that still strongly activates the neuron, with a maximum activation of 0.37 on the word **“native”** in the full example, and a maximum activation of 0.29 on the word **“native”** in the MAE.

Local Search

Dataset examples also often lack diversity and do not illustrate the full range of inputs that are highly activating to a neuron. For example, looking at the same neuron again, many of the examples show the neuron strongly activating to the word **“native”** when used in the context **“I am not a native English speaker”**. This makes it challenging to narrow down to the exact cause of the activation - is the neuron activating to the word **“native”** alone, or only when it occurs as part of the larger phrase? Would **“I am not a native French speaker”** also activate the neuron?

To help answer these questions, we search the local input space around a MAE to find similar inputs, and evaluate whether they still cause a similar activation pattern. To do this, we use BERT [Devlin et al.] to vary each word in turn, by masking the word and taking the top *n* predictions of BERT on the masked input. For each prediction, we substitute it into the sentence

and measure the activation of the target neuron (note that we skip substitution of stop words, as they are uninformative). If the neuron is still activating with less than a *y%* drop-off in activation at the correct position in the sentence, the modified prompt is a new positive example, whereas if the neuron is now activating with more than a *z%* drop-off in activation at the correct position the modified prompt is a negative example. Note that *y* and *z* are parameters which we experimentally set to 40% and 70%. Both positive and negative examples have significant value, as they illustrate what can and cannot change about the sentence to still activate the neuron in the same way. For example, taking the pruned prompt from above, **"am not a native English speaker"**, the positive examples include: **"am not a native language speaker"** and **"am not a native English ."** (the space before the full stop is an artifact of tokenization), whilst the negative examples include **"am not a fluent English speaker"**. This clearly illustrates that the model is responding to the word "native", and does not depend on being followed by "English" or "speaker".

Results

To evaluate the effectiveness of pruning and local search, we performed a case study on a random neuron from each of layer 3 and layer 4 of solu-8L-old. We chose layers 3 and 4 fairly arbitrarily, with the loose intuition that the middle layers would have more interpretable neurons.

Due to time constraints, we unfortunately couldn't extend the case study to more neurons or more layers, although we did do some limited further testing during development and found similar results across layers, although pruning was sometimes unsuccessful on neurons in layers 0 and 1, which seemed more sensitive to the reduced context. More experimentation is definitely needed to have a better understanding of where the method succeeds and fails.

Layer 3, Neuron 912 - see the linked website for the initial dataset examples.

Prompt	**Prompt Type**	**Activation**
Example 0	Dataset Example	1.23
was put on the IUCN Red List under	MAE	0.74
was placed on the IUCN Red List under	Positive	0.68
was listed on the IUCN Red List under	Positive	0.71
was put on the iaaf Red List under	Negative	0.22
was put on the nhl Red List under	Negative	0.20

was put on the IUCN threatened List under	Negative	0.19
was put on the IUCN Red ##lists under	Negative	0.05

Prompt	**Prompt Type**	**Activation**
Example 1	Dataset Example	1.23
It was added to the National Register of	MAE	0.50
It was add to the National Register of	Positive	0.41
It was named to the National Register of	Positive	0.34
It was added to the National list of	Positive	0.44
It was added to the state Register of	Negative	0.13
It was added to the National inventory of	Negative	0.08

These results help to illustrate the function of the neuron more clearly than the dataset examples alone. We can see from both tables that the neuron responds strongly to words like “List” and “Register”, but only in specific contexts. For example, when BERT changes “IUCN” to “iaaf”, the activation decreases significantly, and similarly when BERT changes “National” to “state”. Additionally, the tables show that although the context word (“IUCN” or “National”) is necessary for strong neuron activation on the anchor token, it is not sufficient, as changing the anchor token from “List” to “##lists” or from “Register” to “inventory” significantly reduces activation.

These behaviors are not evident from the dataset examples alone, and demonstrate how exploration in the local input space makes neuron function more interpretable by illustrating the parts of the input that can change more or less freely.

Layer 4, Neuron 912 - see the linked website for the initial dataset examples.

Prompt	**Prompt Type**	**Activation**
Example 0	Dataset Example	2.75
.](bjc201256f1){#fig1	MAE	2.90

.](bjc201256f1)fig1	Positive	2.90
.](bjc201256f1]fig1	Positive	2.90
.](bjc201256f1){#;	Positive	2.90
.](bjc201256f1){#}	Positive	2.90
(bjc201256f1){#fig1	Negative	0.04
#bjc201256f1){#fig1	Negative	0.05
.](\\){#fig1	Negative	0.03
.](thumb){#fig1	Negative	0.02

Prompt	**Prompt Type**	**Activation**
Example 1	Dataset Example	2.75
.](bjc201156f1){#fig1	MAE	2.97
.](bjc201156f1)fig1	Positive	2.65
.](bjc201156f1]fig1	Positive	2.97
.](bjc201156f1]fig1	Positive	2.97
.](bjc201156f1){#}	Positive	2.97
(bjc201156f1){#fig1	Negative	0.05
#bjc201156f1){#fig1	Negative	0.05
.](\\){#fig1	Negative	0.03
.](thumb){#fig1	Negative	0.02

Both tables again show neuron behaviour that is unclear from the dataset examples. By pruning and searching the local input space, we see that the "{#fig1" part of the input is not necessary for neuron activation, and can be changed arbitrarily. On the other hand, the ".]" at the start is essential for strong neuron activation, as is the main string "bjc201156f1". These results are not

intuitive or obvious from just looking at the dataset examples alone, and only become clear due to the local search, which is itself facilitated by the initial pruning.

Conclusions, Limitations, and Future Work

This report describes a pair of methods for investigating neuron activations, which together provide a much clearer picture of neuron function compared to dataset examples alone. By pruning the extraneous information from the long dataset examples to form an MAE, and then searching the local input space around the MAE using contextual word replacement with BERT, the study of neuron behaviour is significantly facilitated, as shown in the case studies where non-obvious behaviour is uncovered. This potentially could be beneficial for interpretability research, as it presents a general and model agnostic mechanism for investigating neuron behaviour that could increase the rate at which researchers can form and test hypotheses.

The primary limitation of the current implementation is its robustness to a variety of neuron behaviours. The implementation is a basic prototype which has been observed to work well for many neurons, but which breaks down in certain situations. For example, the initial pruning to a sentence can be too aggressive and destroy neuron activation. This could be solved by iteratively pruning sentences, before moving on to word level pruning. Additionally, the local search only explores a small section of the search space, as it is limited to replacing a token at a time. Replacing multiple tokens, or using large generative language models to generate entirely different prompts centered around the anchor token, could further illuminate search space and therefore neuron behaviour. Future work could investigate these and other avenues for improving the robustness of the pruning and the power of the local search.

References

See links in main text