# Exploratory Model Analysis Using Data-Driven Neuron Representations

**Daisuke Oba**
The University of Tokyo

**Naoki Yoshinaga**
Institute of Industrial Science,
The University of Tokyo

**Masashi Toyoda**
Institute of Industrial Science,
The University of Tokyo

{oba,ynaga,toyoda}@tkl.iis.u-tokyo.ac.jp

## Abstract

Probing classifiers have been extensively used to inspect whether a model component captures specific linguistic phenomena. This top-down approach is, however, costly when we have no probable hypothesis on the association between the target model component and phenomena. In this study, aiming to provide a flexible, exploratory analysis of a neural model at various levels ranging from individual neurons to the model as a whole, we present a bottom-up approach to inspect the target neural model by using neuron representations obtained from a massive corpus of text. We first feed massive amount of text to the target model and collect sentences that strongly activate each neuron. We then abstract the collected sentences to obtain neuron representations that help us interpret the corresponding neurons; we augment the sentences with linguistic annotations (*e.g.,* part-of-speech tags) and various metadata (*e.g.,* topic and sentiment), and apply pattern mining and clustering techniques to the augmented sentences. We demonstrate the utility of our method by inspecting the pre-trained BERT. Our exploratory analysis reveals that i) specific phrases and domains of text are captured by individual neurons in BERT, ii) a group of neurons simultaneously capture the same linguistic phenomena, and iii) deeper-level layers capture more specific linguistic phenomena.

## 1 Introduction

Deep neural networks (DNNs) learn to induce internal feature representations or neurons[1] for a given input, which are optimized for the target task. The success of DNNs in the field of natural language processing (NLP) is underpinned by this flexibility to induce internal vector representations of input. It is, however, difficult for humans to interpret their roles and properties, which hinders us from leveraging DNNs in practical applications.

[1]**Neurons** refer to numerical values of each dimension of internal representations such as hidden states.

Researchers have therefore investigated what linguistic aspects of input text are captured by the internal representations of DNNs optimized for NLP tasks (§ 2). Most of the existing methods inspect a pre-specified model component (*e.g.,* individual BERT layers) in a top-down manner. A typical approach first takes aim at specific linguistic phenomena that would be captured by the target components, and then trains a probing classifier that predicts the chosen linguistic phenomena from the target components (Bau et al., 2018; Giulianelli et al., 2018; Dalvi et al., 2019; Lakretz et al., 2019; Kovaleva et al., 2019; Goldberg, 2019; Petroni et al., 2019; Hewitt and Manning, 2019; Jawahar et al., 2019; Durrani et al., 2020; Zhou and Srikumar, 2021; Cao et al., 2021; Jumelet et al., 2021).

Although this top-down approach based on probing classifiers has provided thought-provoking insights into the target model components, it becomes costly when we want to inspect many combinations of model components and linguistic phenomena. This is because the probing requires us to train machine-learning classifiers for each combination of model component and linguistic phenomenon. Although there are a few bottom-up approaches to inspect DNNs by examining the response of the neurons towards (perturbed) inputs that represent the target linguistic phenomenon (Karpathy et al., 2016; Shi et al., 2016; Qian et al., 2016), these approaches are labor intensive since they require manual intervention.

This study aims at efficiently inspecting neural NLP models at various levels of granularity ranging from individual neurons to the model as a whole, and presents a bottom-up approach to inspecting what kind of concrete linguistic phenomena each neuron of the model responds to, without presupposing the target linguistic phenomenon to be examined (§ 3). Given a massive amount of text, we first feed individual sentences to the target model, and collect sentences to which each neuron strongly



*Proceedings of the Fourth BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP*, pages 518–528

responds (§ 3.1).[2] We next annotate the collected sentences with the linguistic annotations (*e.g.,* part of speech tags) and various metadata (*e.g.,* topic and sentiment). With the help of these annotations, we then apply text mining techniques such as frequent pattern mining to extract common patterns as *the linguistic signatures*[3] that exist repeatedly and intricately in the sentences collected for the target neurons (§ 3.2). We finally investigate relationships between multiple neurons by comparing and clustering continuous neuron representations induced from the collected sentences.

We apply our method to the pre-trained BERT (base-uncased) (Devlin et al., 2019) to demonstrate how much insight into BERT the method can actually provide. Our exploratory model analysis have confirmed that it is possible to identify, without any prior assumptions, a wide variety of specific linguistic phenomena to which each neuron responds (§ 4.2). Furthermore, by comparing the linguistic phenomena and sentences corresponding to individual neurons, we revealed the existence of neurons that work cooperatively for the same purpose. We finally revealed the impact of optimizing BERT to the target task (here, the pre-training task and sentiment classification) by comparing neurons of randomly initialized BERT with neurons of the pre-trained and the fine-tuned BERT (§ 4.3).

The contribution of this study is threefold:

- We present a method of **exploratory model analysis to understand neural NLP models**. We investigate concrete linguistic phenomena (*e.g.,* skip n-grams) captured by the neurons of the target model, without any prior assumptions about the phenomena to be examined.
- We revealed **concrete linguistic phenomena captured by various BERT's neurons** and layers, whereas existing approaches reported coarse linguistic phenomena that are indicated by high performance in probing tasks.
- We confirmed that **neurons in deeper BERT layers capture more specific linguistic phenomena, and fine-tuning facilitates this tendency** in a sentiment classification task.

[2]For simplicity, we hereafter assume that input to the target model is a single sentence, although our method can be applied to models that takes input in shorter or larger linguistic units.

[3]This term is named after topic signatures (Agirre et al., 2001), which are supplemental data-driven representations of word senses whose representations are extracted from external language resources.

## 2 Related Work

In order to clarify what kind of linguistic phenomena are captured by neural NLP models, researchers have analyzed the internal representations of the models at various levels of granularity such as neurons (Karpathy et al., 2016; Shi et al., 2016; Qian et al., 2016; Bau et al., 2018; Lakretz et al., 2019; Vig et al., 2020; Cao et al., 2021), layers (Hewitt and Manning, 2019; Liu et al., 2019; Tenney et al., 2019a; Goldberg, 2019; Jawahar et al., 2019; Miaschi and Dell'Orletta, 2020), attentions (Kovaleva et al., 2019; Clark et al., 2019; Brunner et al., 2020; Kobayashi et al., 2020), and the model as a whole (Petroni et al., 2019; Broscheit, 2019; Roberts et al., 2020). In what follows, we start by reviewing probing methods that investigate the classifier performance of external tasks based on the target model components. We next describe some microscopic methods that focus on individual neurons. We then discuss a method of inspecting each neuron using text generated for that neuron.

Most of the recent methods adopt a top-down approach called *probing*, which takes target component (*e.g.*, layer) as inputs, trains a classifier that predicts the linguistic phenomena of interest such as syntactic information (Jawahar et al., 2019; Miaschi and Dell'Orletta, 2020; Wu et al., 2020) , agreement information (Giulianelli et al., 2018; Goldberg, 2019), and semantic knowledge (Tenney et al., 2019b; Ettinger, 2020), and evaluates the properties of the internal representation with reference to the accuracy of the classifier. To reduce the cost of training a classifier, Zhou and Srikumar (2021) indirectly predict the performance of probing classifiers by analyzing how the labeled data is represented in the vector space. Some studies identify neurons which make a huge contribution to solving the desired task, by looking at the performance of the task when the activation of neurons is forcibly controlled (Bau et al., 2018; Lakretz et al., 2021; Cao et al., 2021).

These probing methods are inductive in that they examine whether the target model component captures the target linguistic phenomenon. It is costly to inspect various pairs of model components and phenomena. Our approach does not presuppose linguistic phenomena to be examined and gives information that differs from the above methods; we show concrete linguistic phenomena each neuron responds to strongly, in a form that can be easily understood by humans (*e.g.*, skip n-grams) (§ 4.2).

| Datasets | #sent. | domain | metadata | avg. length | distribution of simplified POS tags |
|---|---|---|---|---|---|
| BookCorpus (Zhu et al., 2015) | 40M | book | author | 16.3 | noun: 21.4%, verb: 17.6%, adj.: 5.1% |
| English Wikipedia[4] | 40M | wikipedia | entity | 22.0 | noun: 32.5%, verb: 11.7%, adj.: 6.0% |
| Sentiment140 (Go et al., 2009) | 2M | social media | sentiment | 10.3 | noun: 23.4%, verb: 18.4%, adj.: 5.8% |
| IMDB (Maas et al., 2011) | 0.3M | review | sentiment | 23.8 | noun: 22.9%, verb: 15.3%, adj.: 7.4% |
| 20Newsgroups (Lang, 1995) | 0.2M | news | topic | 23.6 | noun: 29.0%, verb: 13.0%, adj.: 5.3% |
| Reuters[5] | 39K | news | category | 28.9 | noun: 38.2%, verb: 11.7%, adj.: 6.2% |
| Total | 82M | - | - | 18.9 | noun: 27.7%, verb: 14.2%, adj.: 5.7% |

Table 1: Datasets used to collect sentences to which individual neurons of the target model strongly activate. Among the PTB annotations obtained by the POS tagger described in § 4.1, we here defined those POSs whose letters begin with N, V, and J as "noun," "verb," and "adj," respectively.

There are several studies that have investigated the roles of neurons – the finest components of models such as a cell state in a long short term memory – by observing interactions between their values and (perturbed) input sentences (Karpathy et al., 2016; Shi et al., 2016; Qian et al., 2016; Vig et al., 2020; Lakretz et al., 2021). However, these methods are tailored to analyze a few pre-defined particular phenomena such as sequence length, or need to perform the analysis per phenomenon. Our method, on the other hand, allows for a wide range of analytical perspectives in that it uses a massive amount of raw text to collect sentences in which individual neurons show strong interest and leverages pattern mining to highlight linguistic phenomena contained in the collected sentences.

As the most similar to our approach, Poerner et al. (2018) proposed a gradient-based method that generates input text which strongly activates neurons, inspired by work in computer vision (Simonyan et al., 2014). Their method differs from the probing methods in that it embodies the information captured by the neurons as text, which is similar to our study. However, their method can generate only text of pre-fixed length, and requires hyper-parameter tuning (*e.g.,* annealing temperature) to generate natural text, making it costly to apply to massive neurons in the models. Moreover, it is difficult for humans to interpret properties of the generated text just by observing them. In contrast, we propose an example-based method for collecting human-written sentences from a huge text corpora, determining which each neuron finds interesting, without any parameter tuning. By exploiting text mining techniques, we can reveal various linguistic phenomena as long as they appear in the source raw text.

[4]12/20/2020 ver. https://dumps.wikimedia.org/enwiki

[5]http://kdd.ics.uci.edu/databases/reuters21578

## 3 Data-Driven Inspection of Neurons

This section describes our methodology for providing a deeper insight into a neural model at the finest level of granularity (i.e., neurons[1]). Our approach abstracts the sentences retrieved from a massive amount of text such as the BookCorpus (Zhu et al., 2015). The whole process is as follows.

**Step 1: Collecting sentences activating neurons** This step finds which neurons consider what text to be interesting. Feeding a massive amount of raw text such as Wikipedia and BookCorpus (Zhu et al., 2015) to the target model, we collect a large number of sentences that are strongly associated with each neuron in the model (§ 3.1).

**Step 2: Abstracting the collected sentences** This step finds what concrete linguistic phenomena each neuron finds interesting. We associate the collected sentences with linguistic annotations such as part-of-speech tags and available metadata such as topic and sentiment (Table 1). We then abstract a wide range of linguistic phenomena that exist in the retrieved sentences using data mining techniques such as frequent pattern mining, utilizing the associated information (§ 3.2). We call the resulting patterns as *linguistic signatures* of neurons.

### 3.1 Collecting Sentences Activating Neurons

We first feed a massive amount of text to the target model, and then extract sentences which strongly activate each neuron. In other words, we look for sentences that each neuron finds interesting. For this purpose, the size of the corpus can be reduced if there is a large diversity in the linguistic phenomena and domains contained in the corpus.

Since it is impractical to record the neurons' values for all the sentences, we use a priority queues to maintain only the top-$N$ sentences that strongly activate each neuron.

### 3.2 Abstracting the Collected Sentences

Although the collected sentences embody various linguistic phenomena, it is difficult to induce the types of covered linguistic phenomena due to the diversity and redundancy of language. We therefore exploit data-mining techniques such as frequent pattern mining to obtain common patterns as *linguistic signatures*, which represent abstract properties of the sentences, for neurons. To help the pattern mining sum up sentences, we augment the collected sentences with linguistic annotations (here, part-of-speech tags) and metadata in advance.

**Single-neuron analysis** To understand which types of phenomena are captured by each neuron, we first rely on frequent (skip) $n$-grams obtained by using sequential pattern mining (Han et al., 2001).[6] We used relative frequency to find n-grams specific to individual neurons. We can also observe the distribution of the linguistic annotations (*e.g.,* part-of-speech tags) and metadata which capture traits of the sentences (*e.g.,* domain). By encoding the sentences collected for each neuron into vectors and clustering them, we can find the typical sentence which strongly activates the target neuron.

**Cross-neuron analysis** We compare the linguistic signatures (*e.g.*, common $n$-grams and distribution of metadata) obtained for individual neurons. We also extract groups of neurons that work together to capture the same linguistic phenomena by clustering continous neuron representations induced from the sentences collected for the neurons.

## 4 Experiments

Taking the pre-trained BERT (base-uncased) (Devlin et al., 2019) as an example, we demonstrate that our methodology can perform similar analysis to that obtained by existing approaches. We also provide novel findings on how and what aspects of language each neuron captures.

### 4.1 Settings

**Data** We use six English corpora in various domains to mine neuron-sentence interactions (Table 1). We split each corpus into sentences with a sentence tokenizer[7], and normalize repetitions of symbols[8] to a single symbol (*e.g.*, from +++ to +). We then tokenize the resulting sentences with FastTokenizer,[9] and exclude sentences less than three tokens in length. In addition to metadata associated with text in the corpora, we annotate the sentences with part-of-speech (POS) tags using a POS tagger.[10] The statistics of the resulting corpora are summarized in Table 1.

**Models** We adopt the pre-trained BERT (base-uncased, 12 layers, and 768 dimensions of hidden states)[11] as the target for inspection (**pre-trained**). To investigate the effect of representation learning (*e.g.*, masked language modeling and fine-tuning) on the properties of neurons, we also examine two models: one with parameters randomly initialized (**random**) and the other with parameters fine-tuned on sentiment classification task (**fine-tuned**).

To fine-tune BERT, we use the training set in Sentiment140 (Go et al., 2009) that annotates the sentences with polarity labels (positive and negative). We passed the CLS token's output vector through a fully-connected layer, and updated the parameters of pre-trained with a learning rate of $10^{-5}$, dropout rate of $0.3$, batch size of $32$, and $5$ epochs. To obtain the **random** model, we randomly re-initialized only the parameters of the **pre-trained** model other than the word embedding layer since this paper conducts the analysis of 12-layer encoder's neurons excluding the embedding layers.

In experiments, we treat each hidden state as a neuron to be analyzed (i.e., $12 \times 768$ neurons in total in each model). Here we take the average over the hidden states of each token in the sentence for quantifying how strongly the neuron responds to the input. This is to find interesting "sentences" first, and then to find interesting words and phrases exploiting well-established pattern mining methods as described in § 3. Interesting future work will associate with each token the value of the hidden states after processing that token, and use the values to amplify the frequency of n-grams.

To find typical sentences among the sentences collected for each neuron by clustering the 10K sentences for each neuron (single-neuron analysis) and to find a group of neurons that capture similar linguistic phenomena by clustering continuous neuron representations induced from the 10K sentences (cross-neuron analysis), we represent each

[6] Although here we use only part-of-speech taggers for annotation, we can exploit syntactic parsers and FREQT (Asai et al., 2004) to obtain common syntactic structures.

[7] nltk sentence tokenizer ver. 3.2.4

[8] !@#$%^&∗()_+-=[]{};:?,.

[9] https://github.com/huggingface/tokenizers

[10] https://www.logos.ic.i.u-tokyo.ac.jp/ tsuruoka/lapos

[11] https://github.com/huggingface/pytorch-pretrained-BERT

| Noun neuron; 740th hidden state in 3rd layer | |
|---|---|
| words | william, john, roBERT, de, edward |
| word pattern | (Sir William), (Sir John) |
| POS pattern | (NNP NNP NNP), (NNP IN NNP), (NNP NNP IN) |
| ratio of POS | noun: 91.7%, verb: 0.9%, adj.: 0.4% |
| avg. length | 3.8 |
| typical sent. | *The architects of the Square Mile included RoBERT Findlay, Bruce ...* |

| Verb neuron; 462nd hidden state in 6th layer | |
|---|---|
| words | going, got, help, need, get, know |
| word pattern | (going to have), (we have to) |
| POS pattern | (PRP TO VB), (VBP TO VB), (VBG TO VB) |
| ratio of POS | noun: 16.6%, verb: 29.8%, adj.: 1.1% |
| avg. length | 6.4 |
| typical sent. | *Something was going to have to give, and it wasn't going to be him.* |

| Social Media neuron; 200th hidden state in 6th layer | |
|---|---|
| Words | lol, im, u, twitter, haha, ur, like, miss |
| word pattern | (I my), (i it), (i i), (I was), (i lol) |
| POS pattern | (NN NN NN), (NN NN), (PRP NN) |
| ratio of POS | noun: 29.8%, verb: 18.3%, adj.: 6.3% |
| avg. length | 7.6 |
| typical sent. | *nooo I got so many tho That sucks I finished the boook and now I'm ...* |

| Short sentence neuron; 169th hidden state in 1st layer | |
|---|---|
| words | war, party, revolutionary, ibn |
| word pattern | (of the), (the of), (Giovanni Battista) |
| POS pattern | (NNP NNP NNP), (NNP IN NNP), (DT NNP NNP) |
| ratio of POS | noun: 67.5%, verb: 5.5%, adj.: 5.3% |
| avg. length | 3.5 |
| typical sent. | *The Tale of Loyal Knights and ...* |

| Science neuron; 705th hidden state in 9th layer | |
|---|---|
| words | species, family, genus, found, sea |
| word pattern | (is in the), (is of the), (is a the) |
| POS pattern | (IN NNP), (DT IN), (IN DT) |
| ratio of POS | noun: 34.4%, verb: 11.5%, adj.: 7.0% |
| avg. length | 16.7 |
| typical sent. | *Pachliopta polyphontes is a species of butterfly from the family Papilionidae...* |

| Positive neuron; 43rd hidden state in 7th layer | |
|---|---|
| words | chapter, love, welcome, good, ha |
| word pattern | (Chapter Chapter Chapter), (the of the) |
| POS pattern | (DT JJ NN), (PRP DT NN), (PRP RB JJ) |
| ratio of POS | noun: 42.3%, verb: 7.7%, adj.: 4.2% |
| avg. length | 8.1 |
| typical sent. | *doing great sweetie, about to take my dog for a walk in the park (while ...* |

| United States neuron; 203rd hidden state in 7th layer | |
|---|---|
| words | freyja, united, states, house, election |
| word pattern | (is going be), (was going be) |
| POS pattern | (DT JJ NN), (PRP DT NN), (PRP RB JJ) |
| ratio of POS | noun: 63.3%, verb: 1.6%, adj.: 2.8% |
| avg. length | 8.0 |
| typical sent. | *1912 United States House of Representatives election in Wyoming ...* |

| Olympic neuron; 69th hidden state in 6th layer | |
|---|---|
| words | olympics, summer, games |
| word pattern | (United States in), (United States of) |
| POS pattern | (NNP NNP NNP), (NNP IN NNP) |
| ratio of POS | noun: 36.2%, verb: 7.4%, adj.: 3.6% |
| avg. length | 17.4 |
| typical sent. | *In the 5000 metres she competed at the 1995 World Championships and the ...* |

Table 2: Examples of neurons and their linguistic signatures extracted from the top-10K sentences to which they strongly responded. The neurons were manually named to highlight the linguistic phenomena they capture (§ 4.2). The "word" and "word/POS pattern" rows show the most frequent patterns in the 10K sentences. The "typical sent." is that is closest to the average vector of the collected 10K sentences when they are vectorized (§ 4.1).

sentence as vectors, and take average over the 10K sentence vectors to obtain neuron representations. Specifically, we represent each sentence with an average of 300-dimensional fastText embeddings (Bojanowski et al., 2017)[12] of the tokens.

### 4.2 Single-Neuron Analysis

We collected 10K sentences corresponding to each neuron, and identified the common linguistic phenomena in those sentences by using frequent pattern mining. We found interpretable neurons from several perspectives. As inspired by Quiroga et al. (2005) that reports the existence of a "Halle Berry neuron" in a human brain analysis, we gave distinguished names to represent the roles of neurons (*e.g., Science neuron* is activated by text in science domain), and showed some of the linguistic phenomena they captured in Table 2.

With the help of the automatic linguistic annotations, we found neurons that responded only to nouns or verb-rich sentences (*Noun* or *Verb neuron*), and neurons that responded only to short sentences (*Short-length neuron*). Digging deeper into what kind of text *Noun neuron* is actually responding to, we can see another aspect; it responds to names of people such as "william" and "john." It would be very interesting to investigate the relationship between the existence of neurons corresponding to these fundamental linguistic annotations and the task performance of the model.

By looking at the metadata distribution for the collected sentences, we also found neurons that responded to specific domains/concepts. For example, 82.2% of the 10K sentences corresponding to one neuron were taken from the social media domain in Table 1 (*Social Media Neuron*), 80.2%

[12] https://github.com/facebookresearch/fastText

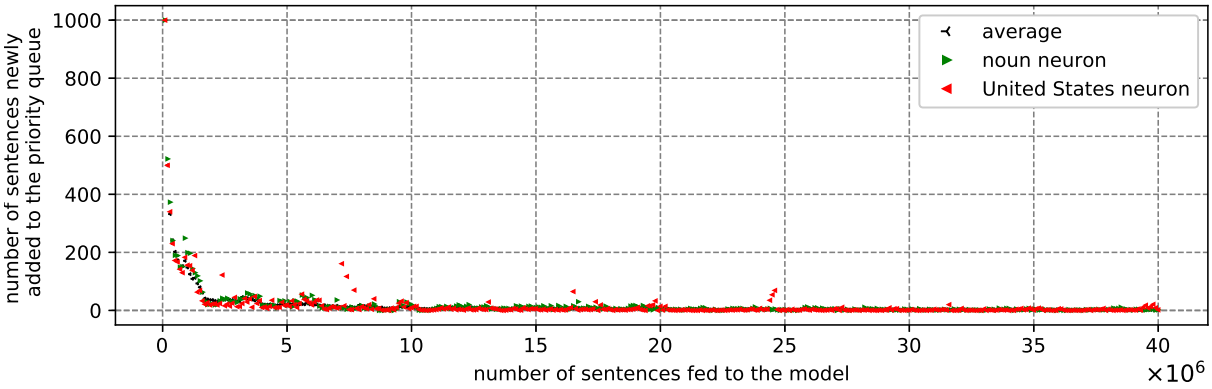



Figure 1: The average number of sentences that were newly added to top-1000 of the priority queues of each neuron for every 100K input sentences when we apply our method to the sentences taken from Wikipedia (Table 1).

of the 10K sentences for another neuron consist of positive sentiment documents[13] (*Positive Neuron*), and 45.3% of the 10K sentences for a third neuron are scientific documents[14] (*Science Neuron*). Moreover, digging deeper into the abstracted linguistic phenomena in Table 2, we can see that *the Social Media neuron* actually responds to informal words such as “lol” and “haha,” and *Science neuron* has a role in responding strongly to relatively uncommon words such as “species” and “genus.” It is an interesting direction to investigate the relationship between the robustness of the model to the domain and the existence of these domain-specific neurons.

We can also reveal the topic-specific neurons by using a list of words that represent particular topics. For example, we have found neurons that are strongly activated by the words “united states, us, u.s., u.s.a, america" (*United States neuron*) and neurons that strongly respond to the words "olympic" (*Olympic neuron*). It would be interesting to investigate the relationship between the existence of such topic-specific neurons and the knowledge possessed by the model (Petroni et al., 2019).

As we have demonstrated so far, the proposed method can find characteristic neurons from various perspectives in a bottom-up manner. It can also reveal information about the properties of neurons (i.e., specific text and linguistic phenomena) that are not covered by the probing task performance (i.e., numerical value). By comparing neurons based on the collected sentences and the linguistic signatures (abstracted phenomena), we can gain insight into the target model components at any level of granularity, as we will demonstrate in the next section.

[13]Sentences of *positive* class in Sentiment140 dataset.
[14]Sentences of *sci* class in 20Newsgroups dataset.

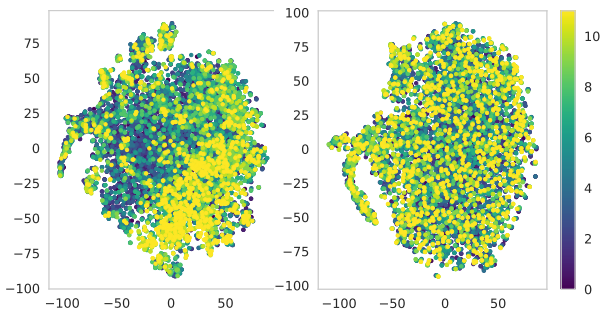
**On the frequency in updating top-$N$ sentences** With the datasets in Table 1 and our server with two Intel® Xeon® E5-2680 v4 2.40-GHz CPUs and eight NVIDIA Quadro P6000 GPUs, analysis of BERT by our method took about 10 hours for Step 1, and about five minutes per neuron for Step 2. The computation time in Step 1 can be reduced if we stop feeding sentences to the target model when the elements of priority cues for neurons become stable.

To confirm this, we performed the Step 1 using the 40M sentences in the Wikipedia corpus in Table 1 with a single GPU on the same server, and checked how many sentences were newly added to the priority queues on a neuron-average basis for every 100K sentences input. Figure 1 shows that after feeding about 2M sentences into the model, 96% of the sentences in the priority queues have already been fixed. This indicates that we can truncate the computation in Step 1 by monitoring the frequency in updates on the priority cues.

We should mention that priority queues for a few neurons are frequently updated even after feeding the model with millions of sentences. This happens if the neurons capture very specific phenomenon, such as *the United States neuron* in Table 2. This suggests that we should diversify the feeding sentences to include various linguistic phenomena as possible when we want to reduce the amount of the corpus fed to the model.

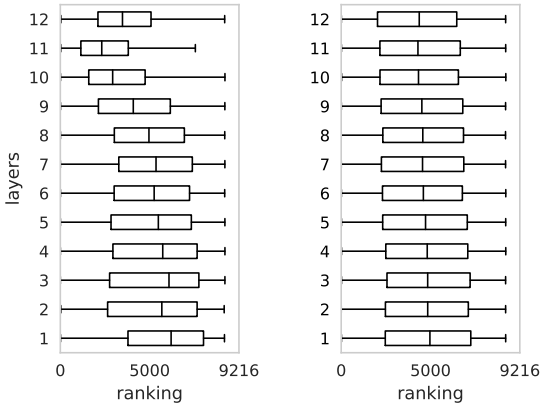
### 4.3 Cross-Neuron Analysis

**Clustering neuron vectors induced from the collected sentences** We leverage the collected 10K sentences for individual neurons to obtain example-based neuron representations (vectors), and then cluster the obtained neuron vectors to identify co-



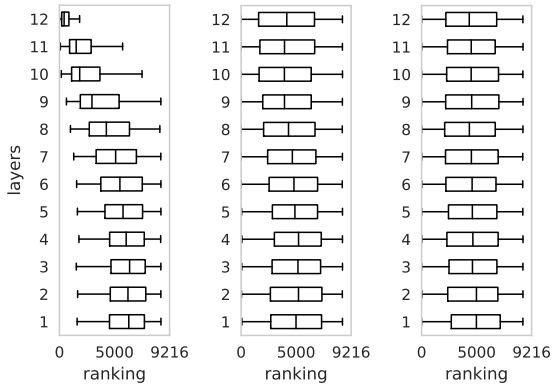
(a) pre-trained (b) random

Figure 2: Clustering neurons in both models. Each neuron is represented by the average of the sentence vectors of the corresponding 10K sentences. The color indicates the layer in which the neuron is located.




(a) pre-trained (b) random

Figure 3: Ranking neurons in the models in terms of how selectively the neuron responds to a specific part-of-speech. The box-and-whisker diagrams show where the neurons in each layer are distributed in the ranking.




(a) fine-tuned (b) pre-trained (c) random

Figure 4: Ranking neurons by standard deviation of sentiment polarity of the collected sentences. The box-and-whisker diagrams show the distribution of each layer's neurons in the ranking.

operating neurons. Here we verify a hypothesis that neurons in the same layer are cooperating. First, we represent each neuron with an average of the vectors for the corresponding 10K sentences. Next, we perform $k$-means clustering of these $12 \times 768$ neuron vectors. The number of clusters is set to 12, which is identical to the number of layers. We report the accuracy of the clustering using the Hungarian algorithm (Kuhn, 1955).

The clustering accuracy is 15.9% for the **pre-trained** model compared to 9.6% for the **random** model. This result suggests that although the overall performance is low, the pre-training made the neurons in each layer capture similar linguistic phenomena, reconfirming results of the existing studies that explore each layer in a top-down manner (§ 2) had a reasonable point of view.

Figure 2 visualizes the neuron vectors using t-SNE (van der Maaten and Hinton, 2008). The heatmap shows which layer the neurons (vectors) belong to. We can see that the pre-trained BERT's neurons in the shallower and deeper layers form different clusters.

**Ranking neurons by the skewness of captured linguistic phenomena** By ranking (sorting) the neurons by the frequency of each linguistic phenomenon and the deviation of their distribution, we can examine where the neurons that represent each linguistic phenomenon are distributed in the model. Here, we take the simplified version of the PTB part-of-speech tags for nouns, verbs, and adjectives (Table 1) and binary sentiment polarity as the target linguistic phenomenon.

For part-of-speech tags, we find neurons that selectively respond to a specific part-of-speech tag. First, we normalize the frequency of each part of speech tag in the 10K sentences collected for each neuron using the frequency obtained from the original corpus (Table 1). The maximum relative frequency (percentage) among all the part-of-speech tags is then used as a measure of how selectively the neuron responds to a particular part of speech tag. For sentiment polarity, we find the neurons that selectively respond to a particular polarity (i.e., positive and negative). First, we calculate the standard deviation of the binary sentiment polarity distribution of the 10K sentences collected for each neuron, and then use this as a measure of how much the neuron responds to a particular sentiment.

Figure 3 and 4 show the results of ranking the neurons in each layer for their selective responsive-

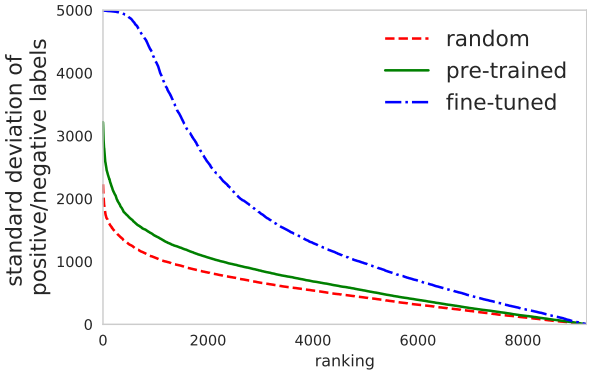



Figure 5: Ranking neurons in the three models by standard deviation of sentiment polarity of the collected sentences.

ness to part-of-speech and sentiment polarity. The box-and-whisker diagrams show where the neurons in each layer are distributed in the ranking. From the figures, we can see that **random** model results in a ranking of neurons that is independent of the layer, while the **pre-trained** model tends to show that neurons in deeper layers are more selectively responsive to both phenomenon.

Interestingly, for the POS tagging task, it has been reported that neurons in the lower layers are better able to handle the probing task (Tenney et al., 2019a). This suggests that, for POS tagging, the presence of neurons that respond uniformly to part-of-speech tags has a significant influence on POS tagging performance. For sentiment polarity, we also calculated the ranking of the neurons in the **fine-tuned** model. Figure 4 shows that the neurons of the **fine-tuned** model in the deeper layers become more capable of capturing the sentiment of the text, while the neurons in the shallower layers are kept insensitive to the polarity of the text.

Figure 5 depicts the results of ranking the neurons in each model, where the y-axis represents the standard deviation of sentiment polarity of the collected sentences for the ranked neurons. We can see that the models have increased the percentage of neurons that show a certain degree of selective response to specific sentiment polarity by pre-training and by fine-tuning. It will be interesting to utilize this standard deviation to measure the distance between the two tasks.

## 5 Conclusions

This study aims to provide a method of exploratory model analysis for a neural NLP model at various levels of granularity ranging from individual neurons to the model as a whole, and proposed a bottom-up methodology for revealing what kind of concrete linguistic phenomena each neuron of the model strongly responds to. We take advantage of large-scale text data and data mining techniques to extract linguistic signatures (common patterns) that characterize the individual neurons.

Taking BERT (base-uncased) as an example, we first showed that specific phrases such as those related to United States are captured by individual neurons in BERT (*e.g., United States Neuron* in Table 2). By comparing neurons in terms of the collected sentences they strongly respond to, we then revealed that neurons in the same layer of BERT have similar properties. Lastly, in comparing the corresponding linguistic phenomena with those of neurons in randomly initialized and fine-tuned models, we found that neurons in deeper BERT layers capture more linguistic phenomena specific to language modeling and sentiment classification.

In the future, we plan to investigate models with different architectures. For example, our method can be used to compare the differences of encoder and decoder neurons in end2end models; there is a debate in the field of neural machine translation as to whether modifying the encoder or the decoder contributes more to domain adaptation (Wang et al., 2021).

Our method enables us to find non-apriori linguistic phenomena the neurons may capture, so that it is possible to assist in the construction of a novel training/evaluation dataset for the probing-based evaluation methods. In addition, our method provides information of a different nature, i.e., the concrete linguistic phenomena to which the internal representations respond, as opposed to numerical information such as accuracy of probing tasks.

As mentioned in § 4.2, our method has the potential to perform with less time complexity by reducing the corpus size fed to the target model in Step 1. Therefore, we plan to study how to select a text corpus of such a small size that it reproduces the analysis results obtained when using a very large corpus as the one used in this study.

## Acknowledgements

This work was supported by JST CREST Grant Number JPMJCR19A4 including AIP challenge program, Japan and JSPS KAKENHI Grant Number 21H03494. We thank Joshua Tanner and the anonymous reviewers for their valuable comments.

## References


Eneko Agirre, Olatz Ansa, David Martinez, and Eduard Hovy. 2001. Enriching WordNet concepts with topic signatures. In *Proceedings of the NAACL workshop on WordNet and Other Lexical Resources: Applications, Extensions and Customizations*.

Tatsuya Asai, Kenji Abe, Shinji Kawasoe, Hiroshi Sakamoto, Hiroki Arimura, and Setsuo Arikawa. 2004. Efficient substructure discovery from large semi-structured data. *IEICE TRANSACTIONS on Information and Systems*, 87(12):2754–2763.

Anthony Bau, Yonatan Belinkov, Hassan Sajjad, Nadir Durrani, Fahim Dalvi, and James Glass. 2018. Identifying and controlling important neurons in neural machine translation. In *Proceedings of the sixth International Conference on Learning Representations*.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146.

Samuel Broscheit. 2019. Investigating entity knowledge in BERT with simple neural end-to-end entity linking. In *Proceedings of the 23rd Conference on Computational Natural Language Learning*, pages 677–685.

Gino Brunner, Yang Liu, Damian Pascual, Oliver Richter, Massimiliano Ciaramita, and Roger Wattenhofer. 2020. On identifiability in transformers. In *Proceedings of the eighth International Conference on Learning Representations*.

Steven Cao, Victor Sanh, and Alexander Rush. 2021. Low-complexity probing via finding subnetworks. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 960–966.

Kevin Clark, Urvashi Khandelwal, Omer Levy, and Christopher D. Manning. 2019. What does BERT look at? an analysis of BERT's attention. In *Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 276–286.

Fahim Dalvi, Nadir Durrani, Hassan Sajjad, Yonatan Belinkov, Anthony Bau, and James Glass. 2019. What is one grain of sand in the desert? Analyzing individual neurons in deep NLP models. In *Proceedings of the 33th AAAI Conference on Artificial Intelligence*, pages 6309–6317.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1*, pages 4171–4186.

Nadir Durrani, Hassan Sajjad, Fahim Dalvi, and Yonatan Belinkov. 2020. Analyzing individual neurons in pre-trained language models. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*, pages 4865–4880.

Allyson Ettinger. 2020. What BERT is not: Lessons from a new suite of psycholinguistic diagnostics for language models. *Transactions of the Association for Computational Linguistics*, 8:34–48.

Mario Giulianelli, Jack Harding, Florian Mohnert, Dieuwke Hupkes, and Willem Zuidema. 2018. Under the hood: Using diagnostic classifiers to investigate and improve how language models track agreement information. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 240–248.

Alec Go, Richa Bhayani, and Lei Huang. 2009. Twitter sentiment classification using distant supervision. *CS224N project report, Stanford*, 1(12).

Yoav Goldberg. 2019. Assessing BERT's syntactic abilities. *arXiv preprint arXiv:1901.05287*.

Jiawei Han, Jian Pei, Behzad Mortazavi-Asl, Helen Pinto, Qiming Chen, Umeshwar Dayal, and Meichun Hsu. 2001. Prefixspan: Mining sequential patterns efficiently by prefix-projected pattern growth. In *Proceedings of the 17th International Conference on Data Engineering*, pages 215–224.

John Hewitt and Christopher D. Manning. 2019. A structural probe for finding syntax in word representations. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1*, pages 4129–4138.

Ganesh Jawahar, Benoît Sagot, and Djamé Seddah. 2019. What does BERT learn about the structure of language? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3651–3657.

Jaap Jumelet, Milica Denic, Jakub Szymanik, Dieuwke Hupkes, and Shane Steinert-Threlkeld. 2021. Language models use monotonicity to assess NPI licensing. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 4958–4969.

Andrej Karpathy, Justin Johnson, and Li Fei-Fei. 2016. Visualizing and understanding recurrent networks. In *Proceedings of the fourth International Conference on Learning Representations – Workshop Track*.

Goro Kobayashi, Tatsuki Kuribayashi, Sho Yokoi, and Kentaro Inui. 2020. Attention is not only a weight: Analyzing transformers with vector norms. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*, pages 7057–7075.

Olga Kovaleva, Alexey Romanov, Anna Rogers, and Anna Rumshisky. 2019. Revealing the dark secrets of BERT. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, pages 4365–4374.

Harold W Kuhn. 1955. The Hungarian method for the assignment problem. *Naval research logistics quarterly*, 2(1-2):83–97.

Yair Lakretz, Dieuwke Hupkes, Alessandra Vergallito, Marco Marelli, Marco Baroni, and Stanislas Dehaene. 2021. Mechanisms for handling nested dependencies in neural-network language models and humans. *Cognition*, page 104699.

Yair Lakretz, Germán Kruszewski, Théo Desbordes, Dieuwke Hupkes, Stanislas Dehaene, and Marco Baroni. 2019. The emergence of number and syntax units in lstm language models. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1*, pages 11–20.

Ken Lang. 1995. Newsweeder: Learning to filter netnews. In *Proceedings of the Twelfth International Conference on Machine Learning*, pages 331–339.

Nelson F. Liu, Matt Gardner, Yonatan Belinkov, Matthew E. Peters, and Noah A. Smith. 2019. Linguistic knowledge and transferability of contextual representations. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1*, pages 1073–1094.

Andrew L. Maas, Raymond E. Daly, Peter T. Pham, Dan Huang, Andrew Y. Ng, and Christopher Potts. 2011. Learning word vectors for sentiment analysis. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 142–150.

Alessio Miaschi and Felice Dell'Orletta. 2020. Contextual and non-contextual word embeddings: an in-depth linguistic investigation. In *Proceedings of the 5th Workshop on Representation Learning for NLP*, pages 110–119.

Fabio Petroni, Tim Rocktäschel, Sebastian Riedel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, and Alexander Miller. 2019. Language models as knowledge bases? In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, pages 2463–2473.

Nina Poerner, Benjamin Roth, and Hinrich Schütze. 2018. Interpretable textual neuron representations for NLP. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 325–327.

Peng Qian, Xipeng Qiu, and Xuanjing Huang. 2016. Analyzing linguistic knowledge in sequential model of sentence. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 826–835.

R. Quian Quiroga, Leila Reddy, Gabriel Kreiman, Christof Koch, and Itzhak Fried. 2005. Invariant visual representation by single neurons in the human brain. *Nature*, 435(7045):1102–1107.

Adam Roberts, Colin Raffel, and Noam Shazeer. 2020. How much knowledge can you pack into the parameters of a language model? In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*, pages 5418–5426.

Xing Shi, Kevin Knight, and Deniz Yuret. 2016. Why neural translations are the right length. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2278–2282.

Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. 2014. Deep inside convolutional networks: Visualising image classification models and saliency maps. In *Proceedings of the second the International Conference on Learning Representations – Workshop Track*.

Ian Tenney, Dipanjan Das, and Ellie Pavlick. 2019a. BERT rediscovers the classical NLP pipeline. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4593–4601.

Ian Tenney, Patrick Xia, Berlin Chen, Alex Wang, Adam Poliak, R. Thomas McCoy, Najoung Kim, Benjamin Van Durme, Samuel R. Bowman, Dipanjan Das, and Ellie Pavlick. 2019b. What do you learn from context? Probing for sentence structure in contextualized word representations. In *Proceedings of the seventh International Conference on Learning Representations*.

Laurens van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-SNE. *Journal of Machine Learning Research*, 9:2579–2605.

Jesse Vig, Sebastian Gehrmann, Yonatan Belinkov, Sharon Qian, Daniel Nevo, Yaron Singer, and Stuart M Shieber. 2020. Investigating gender bias in language models using causal mediation analysis. In *Advances in Neural Information Processing Systems 33*, pages 12388–12401.

Yue Wang, Cuong Hoang, and Marcello Federico. 2021. Towards modeling the style of translators in neural machine translation. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1193–1199.

Zhiyong Wu, Yun Chen, Ben Kao, and Qun Liu. 2020. Perturbed masking: Parameter-free probing for analyzing and interpreting BERT. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4166–4176.

Yichu Zhou and Vivek Srikumar. 2021. DirectProbe: Studying representations without classifiers. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5070–5083.

Yukun Zhu, Ryan Kiros, Rich Zemel, Ruslan Salakhutdinov, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. 2015. Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 19–27.