Neuron-level Interpretation of Deep NLP Models: A Survey

Hassan Sajjad[♣*] **Nadir Durrani**[♠*] **Fahim Dalvi**[♠*]
[♣]Faculty of Computer Science, Dalhousie University, Canada[†]
[♠]Qatar Computing Research Institute, HBKU, Doha, Qatar
hsajjad@dal.ca, {ndurrani, faimaduddin}@hbku.edu.qa

Abstract

The proliferation of Deep Neural Networks in various domains has seen an increased need for interpretability of these models. Preliminary work done along this line, and papers that surveyed such, are focused on high-level representation analysis. However, a recent branch of work has concentrated on interpretability at a more granular level of analyzing neurons within these models. In this paper, we survey the work done on neuron analysis including: i) methods to discover and understand neurons in a network; ii) evaluation methods; iii) major findings including cross architectural comparisons that neuron analysis has unraveled; iv) applications of neuron probing such as: controlling the model, domain adaptation, and so forth; and v) a discussion on open issues and future research directions.

1 Introduction

Models trained using Deep Neural Networks (DNNs) have constantly pushed the state-of-the-art in various Natural Language Processing (NLP) problems, for example, Language Modeling (Mikolov et al., 2013; Devlin et al., 2019) and Machine Translation (Sutskever et al., 2014; Bahdanau et al., 2014) to name a few. Despite this remarkable revolution, the black-box nature of DNNs has remained a major bottleneck in their large scale adaptability—especially in the applications where fairness, trust, accountability, reliability, and ethical decision-making are considered critically important metrics or at least as important as the model's performance (Lipton, 2016).

This opaqueness of DNNs has spurred a new area of research to analyze and understand these models. A plethora of papers have been written in the past five years on interpreting deep NLP models and to answer one question in particular: ***What knowledge is learned within representations?*** We term this work as the *Representation Analysis*.

Representation Analysis thrives on post-hoc decomposability, where we analyze the embeddings to uncover linguistic (and non-linguistic) concepts[1] that are captured as the network is trained towards an NLP task (Adi et al., 2016; Belinkov et al., 2017a; Conneau et al., 2018; Liu et al., 2019; Tenney et al., 2019). A majority of the work on *Representation Analysis* has focused on a holistic view of the representations, namely, how much knowledge of a certain concept is learned within representations as a whole (See Belinkov et al. (2020a) for a survey done on this line of work). Recently, a more fine-grained neuron interpretation has started to gain attention. In addition to the holistic view of the representation, *Neuron Analysis* provides insight into a fundamental question: ***How is knowledge structured within these representations?*** In particular, it targets questions such as:

- What concepts are learned within neurons of the network?
- Are there neurons that specialize in learning particular concepts?
- How localized/distributed and redundantly is the knowledge preserved within neurons of the network?

Answers to these questions entail potential benefits beyond understanding the inner workings of models, for example: i) controlling bias and manipulating system's behavior by identifying relevant neurons with respect to a prediction, ii) model distillation by removing less useful neurons, iii) efficient feature selection by selecting

[*]The authors contributed equally.
[†]The work was done while the author was at QCRI.
[1]Please refer to Section 2 for a formal definition.

Transactions of the Association for Computational Linguistics, vol. 10, pp. 1285–1303, 2022. https://doi.org/10.1162/tacl_a_00519
Action Editor: Marco Baroni. Submission batch: 4/2022; Revision batch: 7/2022; Published 11/2022.

Words	Obama	receives	Netanyahu	in	the	capital	of	USA
Suffix	–	s	–	–	–	–	–	–
POS	NNP	VBZ	NNP	IN	DT	NN	IN	NP
SEM	PER	ENS	PER	REL	DEF	REL	REL	GEO
Chunk	B-NP	B-VP	B-NP	B-PP	B-NP	I-NP	B-PP	B-NP
CCG	NP	((S[dcl]\NP)/PP)/NP	NP	PP/NP	NP/N	N	(NP\NP)/NP	NP

Table 1: Example sentences with different word-level concepts. POS: Parts of Speech tags, SEM: Semantic tags, Chunk: Chunking tags, CCG: Combinatory Categorial Grammar tags.

the most salient neurons and removing the redundant ones, and iv) neural architecture search by guiding the search with important neurons.

The work on neuron analysis has explored various directions such as: proposing novel methods to discover concept neurons (Mu and Andreas, 2020; Hennigen et al., 2020), analyzing and comparing architectures using neuron distributions (Wu et al., 2020; Suau et al., 2020; Durrani et al., 2020), and enabling applications of neuron analysis (Bau et al., 2019; Dai et al., 2021). In this survey, we aim to provide a broad perspective of the field with an in-depth coverage of each of these directions. We propose a matrix of seven attributes to compare various neuron analysis methods. Moreover, we discuss the open issues and promising future directions in this area.

The survey is organized as follows: Section 2 defines the terminologies and formally introduces neuron analysis. Section 3 covers various neuron analysis methods and compares them using seven attributes. Section 4 presents the techniques that have been used to evaluate the effectiveness of neuron analysis methods. Section 5 discusses the findings of neuron analysis methods. Lastly, Section 6 showcases various applications of the presented methods and Section 7 touches upon the open issues and future research directions.

2 Definitions

In this section, we define the terminology used in the paper and the objective of neuron analysis more formally.

Neuron Neural networks, such as RNNs or transformer models consist of various components such as gates/cells, blocks, layers, attention heads, and so on. We use the term *neuron* (also called *features*, *experts*, and *units* in the literature) to refer to the output of a single dimension from any neural network component. For example, in the BERT base model, the output of a layer block has 768 neurons and the output of an attention head has 64 neurons. Moreover, we refer to individual neurons that learn a single concept as *focused neurons*, and a set of neurons that in combination represent a concept as *group neurons*.

Concept A concept represents a coherent fragment of knowledge, such as ‘‘a class containing certain objects as elements, where the objects have certain properties’’ (Stock, 2010). For example, a concept could be lexical (e.g., words ending with suffix ‘‘ed’’), morphological (e.g., gerund verbs), or semantic (e.g., names of cities). We loosely define a concept $\mathcal{C}$ as ***a group of words that are coherent with respect to a linguistic property***. Table 1 shows an example sentence with different concept annotations.

Objective Figure 1 presents an overview of various objectives in neuron analysis. Formally, given a model $\mathcal{M}$ and a set of neurons $\mathcal{N}$ (which may consist of all the neurons in the network or a specific subset from particular components like a layer or an attention head) and a concept $\mathcal{C}$, neuron analysis aims to achieve one of the following objectives:

- For a concept $\mathcal{C}$, find a ranked list of $|\mathcal{N}|$ neurons with respect to the concept (dotted blue line)
- Given a neuron $n_i \in \mathcal{N}$, find a set of concepts $|\mathcal{C}|$ the neuron represents (dashed purple line)
- Given a set of neurons, find a subset of neurons that encode similar knowledge (solid green line)

The former two aim to understand what concepts are encoded within the learned representation. The last objective analyzes how knowledge is distributed across neurons. Each neuron $n_i \in \mathcal{N}$

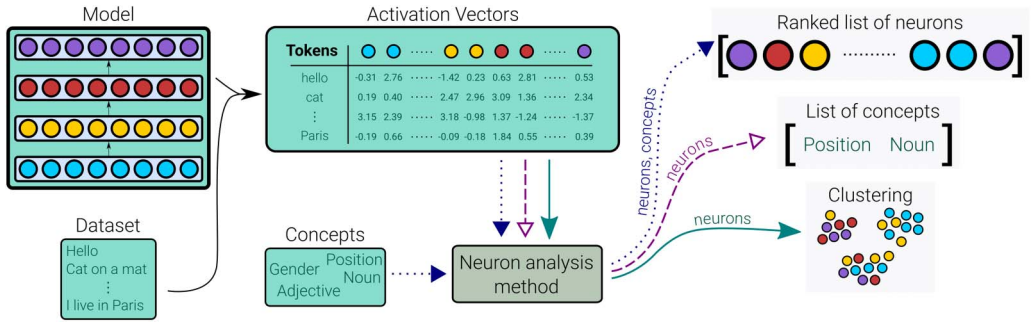


Figure 1: Overview of neuron analysis summarizing the three objectives as discussed in Section 2.

is represented as a vector of activation values over some dataset $\mathcal{D}$. Here, every element of the vector corresponds to a word. For phrase- or sentence-level concepts, an aggregation of neuron activations over words in the phrase/sentence is used. Alternatively, `[CLS]` token representation is also used for transformer models that are transfer learned towards a downstream NLP task.

3 Neuron Analysis Methods

We have classified the work done on neuron analysis into 5 broader categories of methods, namely: i) visualizations, ii) corpus-based, iii) probing-based, iv) causation-based, and v) miscellaneous methods, based on a set of attributes we describe below:

- Scope: Does the method provide global or local interpretation? Global methods accumulate statistics across a set of examples to discover the role of a neuron. Local methods provide interpretation of a neuron in a particular example and may not necessarily reflect its role over a large corpus.
- Input and Output: What is the input (e.g., a set of neurons or concepts) to the method and what does it output?
- Scalability: Can the method be scaled to a larger set of neurons?
- HITL: Does the method require a human-in-the-loop for interpretation?
- Supervision: Does the method depend on labeled data to provide interpretation?
- Causation: Is the interpretation connected with the model's prediction?

Table 2 summarizes and compares each method in the light of these attributes. We discuss them in detail below.[2]

3.1 Visualization

A simple way to discover the role of a neuron is by visualizing its activations and manually identifying the underlying concept over a set of sentences (Karpathy et al., 2015; Fyshe et al., 2015; Li et al., 2016a). Given that deep NLP models are trained using billions of neurons, it is impossible to visualize all the neurons. A number of clues have been used to shortlist the neurons for visualization, for example, selecting saturated neurons, high/low variance neurons, or ignoring dead neurons (Karpathy et al., 2015) when using ReLU activation function.[3]

Limitation While visualization is a simple approach to find an explanation for a neuron, it has some major limitations: i) it is qualitative and subjective, ii) it cannot be scaled to the entire network due to an extensive human-in-the-loop effort, iii) it is difficult to interpret polysemous neurons that acquire multiple roles in different contexts, iv) it is ineffective in identifying *group neurons*, and lastly v) not all neurons are visually interpretable. Visualization nevertheless remains a useful tool when applied in combination to other interpretation methods that are discussed below.

3.2 Corpus-based Methods

Corpus-based methods discover the role of neurons by aggregating statistics over data activations. They establish a connection between a neuron and a concept using co-occurrence between a neuron's

[2]Table 3 in Appendix gives a more comprehensive list.

[3]Saturated neurons have a gradient value of zero. Dead neurons have an activation value of zero.

	Scope	Input	Output	Scalability	HITL	Supervision	Causation
Visualization							
Karpathy et al. (2015)	local	neuron	concept	low	yes	no	no
Corpus-based methods							
Concept Search							
Kádár et al. (2017)	global	neuron	concept	low	yes	no	no
Na et al. (2019)	global	neuron	concept	high	no	no	no
Neuron Search							
Mu and Andreas (2020); Suau et al. (2020); Antverg and Belinkov (2022)	global	concept	neurons	high	no	yes	no
Probing-based methods							
Linear (Dalvi et al., 2019)	global	concept	neurons	high	no	yes	no
Gaussian (Hennigen et al., 2020)	global	concept	neurons	high	no	yes	no
Causation-based methods							
Ablation (Lakretz et al., 2019)	both	concept/ class	neurons	medium	no	no	yes
Knowledge attribution (Dai et al., 2021)	local	concept/ class	neurons	high	no	no	yes
Miscellaneous methods							
Corpus generation (Poerner et al., 2018)	global	neuron	concept	low	yes	no	no
Matrix factorization (Alammar, 2020)	local	neurons	neurons	low	yes	no	no
Clustering (Dalvi et al., 2020)	global	neurons	neurons	high	yes	no	no
Multi model search (Bau et al., 2019)	global	neurons	neurons	high	yes	no	no

Table 2: Comparison of neuron analysis methods based on various attributes. The exhaustive list of citations for each method are provided in the text.

activation values and existence of the concept in the underlying input instances (e.g., word, phrases, or the entire sentence). Corpus-based methods are global interpretation methods as they interpret the role of a neuron over a set of inputs. They can be effectively used in combination with the visualization method to reduce the search space for finding the most relevant portions of data that activates a neuron, thus significantly reducing the human-in-the-loop effort. Corpus-based methods can be broadly classified into two sets: i) the methods that take a neuron as an input and identify the concept the neuron has learned (*Concept Search*), and ii) others that take a concept as input and identify the neurons learning the concept (*Neuron Search*).

Concept Search This set of methods take a neuron as an input and search for a concept that the neuron has learned. They sort the input instances based on the activation values of the given neuron. The top activating instances represent a concept the neuron represents. Kádár et al. (2017) discovered neurons that learn various linguistic concepts using this approach. They extracted top-20, 5-gram contexts for each neuron based on the magnitude of activations and manually identified the underlying concepts. This manual effort of identifying concepts is cumbersome and requires a human-in-the-loop. Na et al. (2019) addressed this by using lexical concepts of various granularities. Instead of 5-gram contexts, they extracted top-k activating sentences for each neuron. They parsed the sentences to create concepts (words and phrases) using the nodes of the parse trees. They then created synthetic sentences that highlight a concept (e.g., a particular word occurring in all synthetic sentences). The neurons that activate largely on these sentences are considered to have learned the concept. This methodology is useful in analyzing neurons that are responsible for multi-word concepts such as

phrases and idiomatic collocations. However, the synthetic sentences are often ungrammatical and lead towards a risk of identifying neurons that exhibit arbitrary behavior (like repetition) instead of concept specific behavior.

Neuron Search The second class of corpus-based methods aim to discover neurons for a given concept. The underlying idea is the same, that is, to establish a link between the concept and neurons based on co-occurrences stats, but in the opposite direction. The activation values play a role in weighing these links to obtained a ranked list of neurons against the concept. Mu and Andreas (2020) achieved this by creating a binary mask of a neuron based on a threshold on its activation values for every sentence in the corpus. Similarly, they created a binary mask for every concept based on its presence or absence in a sentence. They then computed the overlap between a given neuron mask vector and a concept mask vector using intersection-over-union (IoU), and use these to generate compositional explanations. Differently from them, Suau et al. (2020) used the values of neuron activations as prediction scores and computed the average precision per neuron and per concept. Finally, Antverg and Belinkov (2022) considered the mean activation values of a neuron with respect to instances that posses the concept of interest.

The two methods give an alternative view to neuron interpretation. While *Neuron Search* methods aim to find the neuron that has learned a concept, *Concept Search* methods generate explanations for neurons by aligning them with a concept.

Limitation The corpus-based methods do not model the selection of *group neurons* that work together to learn a concept. Concept Search methods consider every neuron independently. Similarly, Neuron Search methods do not find the correlation of a group of neurons with respect to the given concept.

3.3 Probing-based Methods

Probing-based methods train diagnostic classifiers (Hupkes et al., 2018) over activations to identify neurons with respect to predefined concepts. They are a global interpretation methods that discover a set of neurons with respect to each concept using supervised data annotations. They are highly scalable, and can be easily applied on a large set of neurons and over a large set of concepts. In the following, we cover two types of classifiers used for probing.

Linear Classifiers The idea is to train a linear classifier towards the concept of interest, using the activation vectors generated by the model being analyzed. The weights assigned to neurons (features to the classifier) serve as their importance score with respect to the concept. The regularization of the classifier directly effects the weights and therefore the ranking of neurons. Radford et al. (2019) used L1 regularization, which forces the classifier to learn spiky weights, indicating the selection of very few specialized neurons learning a concept, while setting the majority of neurons' weights to zero. Lakretz et al. (2019), on the other hand, used L2 regularization to encourage grouping of features. This translates to discovering *group neurons* that are jointly responsible for a concept. Dalvi et al. (2019) used ElasticNet regularization, which combines the benefits of L1 and L2, accounting for both highly correlated *group neurons* and specific *focused neurons* with respect to a concept.

Limitation A pitfall to probing classifiers is whether a probe faithfully reflects the concept learned within the representation or just memorizes the task (Hewitt and Liang, 2019; Zhang and Bowman, 2018). Researchers have mitigated this pitfall for some analyses by using random initialization of neurons (Dalvi et al., 2019) and control tasks (Durrani et al., 2020) to demonstrate that the knowledge is possessed within the neurons and not due to the probe's capacity for memorization. Another discrepancy in the neuron probing framework, which especially affects the linear classifiers, is that variance patterns in neurons differ strikingly across the layers. Sajjad et al. (2021) suggested applying z-normalization as a pre-processing step to any neuron probing method to alleviate this issue.

Gaussian Classifier Hennigen et al. (2020) trained a generative classifier with the assumption that neurons exhibit a Gaussian distribution. They fit a multivariate Gaussian over all neurons and extracted individual probes for single neurons. A caveat to their approach is that activations do not always follow a *Gaussian prior*

in practice—hence restricting their analysis to only the neurons that satisfy this criteria. Moreover, the interpretation is limited to single neurons and identifying groups of neurons requires an expensive greedy search.

Limitation In addition to the shortcomings discussed above, a major limitation of probing-based methods is the requirement of supervised data for training the classifier, thus limiting the analysis only to predefined or annotated concepts.

3.4 Causation-based Methods

The methods we have discussed so far are limited to identifying neurons that have learned the encoded concepts. They do not inherently reflect their importance towards the model's performance. Causation-based methods identify neurons with respect to model's prediction.

Ablation The central idea behind ablation is to notice the effect of a neuron on model's performance by varying its value. This is done either by clamping its value to zero or a fixed value and observing the change in network's performance. Ablation has been effectively used to find i) salient neurons with respect to a model (unsupervised), and ii) salient neurons with respect to a particular output class in the network (supervised). The former identifies neurons that incur a large drop in model's performance when ablated (Li et al., 2016a). The latter selects neurons that cause the model to flip its prediction with respect to a certain class (Lakretz et al., 2019). Here, the output class serves as the concept against which we want to find the salient neurons.

Limitation Identifying *group neurons* requires ablating all possible combinations of neurons, which is an NP-hard problem (Binshtok et al., 2007). Several researchers have tried to circumvent this by using leave-one-out estimates (Zintgraf et al., 2017), beam search (Feng et al., 2018), learning end-to-end differentiable prediction model (De Cao et al., 2020), and using correlation clustering to group similar neurons before ablation (Dalvi et al., 2020). Nevertheless, all these approaches are approximations and may incur search errors.

Knowledge Attribution Method Attribution-based methods highlight the importance of input features and neurons with respect to a prediction (Dhamdhere et al., 2018; Lundberg and Lee, 2017; Tran et al., 2018). Dai et al. (2021) used an attribution-based method to identify salient neurons with respect to a relational fact. They hypothesized that factual knowledge is stored in the neurons of the feed-forward neural networks of the transformer model and used integrated gradient (Sundararajan et al., 2017) to identify top neurons that express a relational fact. The work of Dai et al. (2021) shows the applicability of attribution methods in discovering causal neurons with respect to a concept of interest and is a promising research direction.

Limitation The attribution-based methods highlight salient neurons with respect to a prediction. What concepts these salient neurons have learned is unknown. Dai et al. (2021) worked around this by limiting their study to model classes where each class serves as a concept. Attribution-based methods can be enriched by complementing them with other neuron analysis methods such as corpus search that associate salient neurons to a concept.

3.5 Miscellaneous Methods

In this section, we cover a diverse set of methods that do not fit in the above defined categories.

Corpus Generation A large body of neuron analysis methods identify neurons with respect to predefined concepts and the scope of search is only limited to the corpus used to extract the activations. It is possible that a neuron represents a diverse concept that is not featured in the corpus. The *Corpus Generation* method addresses this problem by generating novel sentences that maximize a neuron's activations. These sentences unravel hidden information about a neuron, facilitating the annotator to better describe its role. Corpus generation has been widely explored in Computer Vision. For example, Erhan et al. (2009) used gradient ascent to generate synthetic input images that maximize the activations of a neuron. However, a gradient ascent can not be directly applied in NLP, because of the discrete inputs. Poerner et al. (2018) worked around this problem by using *Gumble Softmax* and showed their method to surpass Concept Search method (Kádár et al., 2017) in interpreting neurons.

Limitation Although the corpus generation method has the benefit of generating novel patterns that explain a neuron beyond the space of the underlying corpus, it often generates nonsensical patterns and sentences that are difficult to analyze in isolation. A thorough evaluation is necessary to know its true potential and efficacy in NLP.

Matrix Factorization The Matrix Factorization (MF) method decomposes a large matrix into a product of smaller matrices of factors, where each factor represents a group of elements performing a similar function. Given a model, the activations of an input sentence form a matrix. MF can be effectively applied to decompose the activation matrix into smaller matrices of factors where each factor consists of a set of neurons that learn a concept. MF is a local interpretation method. It is commonly used in analyzing vision models (Olah et al., 2018). We could not find any research using MF on NLP models. To the best of our knowledge, Alammar (2020) is the only blog post that introduced them in the NLP domain.

Limitation Compared to the previously discussed unsupervised methods, MF has an innate benefit of discovering *group neurons*. However, it is still non-trivial to identify the number of groups (factors) to decompose the activations matrix into. Moreover, the scope of the method is limited to local interpretation.

Clustering Methods Clustering is another effective way to analyze groups of neurons in an unsupervised fashion. The intuition is that if a group of neurons learns a specific concept, then their activations would form a cluster. Meyes et al. (2020) used UMAP (Mclnnes et al., 2020) to project activations to a low dimensional space and performed K-means clustering to group neurons. Dalvi et al. (2020) aimed at identifying redundant neurons in the network. They first computed correlation between neuron activation pairs and used hierarchical clustering to group them. The neurons with highly correlated behavior are clustered together and are considered redundant in the network.

Limitation Similar to the MF method, the number of clusters is a hyperparameter that needs to be predefined or selected empirically. A small number of clusters may result in dissimilar neurons in the same group while a large number of clusters may lead to similar neurons split in different groups.

Multi-model Search Multi-model search is based on the intuition that salient information is shared across the models trained towards a task (i.e., if a concept is important for a task then all models optimized for the task should learn it). The search involves identifying neurons that behave similarly across the models. Bau et al. (2019) used Pearson correlation to compute a similarity score of each neuron of a model with respect to the neurons of other models. They aggregated the correlations for each neuron using several methods with the aim of highlighting different aspects of the model. More specifically, they used *Max Correlation* to capture concepts that emerge strongly in multiple models, *Min Correlation* to select neurons that are correlated with many models though they are not among the top correlated neurons, *Regression Ranking* to find individual neurons whose information is distributed among multiple neurons of other models, and *SVCCA* (Raghu et al., 2017) to capture information that may be distributed in fewer dimensions than the whole representation.

Limitation All the methods discussed in this section require human-in-the-loop to provide explanation for the underlying neurons. They can nevertheless be useful in tandem with the other interpretation methods. For example, Dalvi et al. (2019) intersected the neurons discovered via the probing classifier and the multi-model search to describe salient neurons in the NMT models.

4 Evaluation

In this section, we survey the evaluation methods used to measure the correctness of the neuron analysis methods. Due to the absence of interpretation benchmarks, it is difficult to precisely define ‘‘correctness’’. Evaluation methods in interpretation mostly resonate with the underlying method to discovered salient neurons. For example, visualization methods often require qualitative evaluation via human in the loop, probing methods claim correctness of their rankings using classifier accuracy as a proxy. Antverg and Belinkov (2022) highlighted this discrepancy and

suggested disentangling the analysis methodology from the evaluation framework—for example, by using a principally different evaluation method compared to the underlying neuron analysis method. In the following, we summarize various evaluation methods and their usage in the literature.

4.1 Ablation

While ablation has been used to discover salient neurons for the model, it has also been used to evaluate the efficacy of the selected neurons. More concretely, given a ranked list of neurons (e.g., the output of the probing method), we ablate neurons in the model in the order of their importance and measure the effect on performance. The idea is that removing the top neurons should result in a larger drop in performance compared to randomly selected neurons. Dalvi et al. (2019) and Durrani et al. (2020) used ablation in the probing classifier to demonstrate correctness of their neuron ranking method. Similarly, Bau et al. (2019) showed that ablating the most salient neurons, discovered using multi-model search, in NMT models lead to a much bigger drop in performance as opposed to removing randomly selected neurons.

4.2 Classification Performance

Given salient neurons with respect to a concept, a simple method to evaluate their correctness is to train a classifier using them as features and predict the concept of interest. The performance of the classifier relative to a classifier trained using random neurons and least important neurons is used as a metric to gauge the efficacy of the selected salient neurons. However, it is important to ensure that the probe is truly representing the concepts encoded within the learned representations and not memorizing them during classifier training. Hewitt and Liang (2019) introduced Controlled Tasks Selectivity as a measure to gauge this. Durrani et al. (2020) adapted controlled tasks for neuron-probing to show that their probes indeed reflect the underlying linguistic tasks.

4.3 Information Theoretic Metric

Information theoretic metrics such as mutual information have also been used to interpret representations of deep NLP models (Voita and Titov, 2020; Pimentel et al., 2020). Here, the goal is to measure the amount of information a representation provides about a linguistic properties. Hennigen et al. (2020) used mutual information to evaluate the effectiveness of their Gaussian-based method by calculating the mutual information between subset of neurons and linguistic concepts.

4.4 Concept Selectivity

Another evaluation method derived from *Concept Search* methodology measures the alignment between neurons and the discovered concept, by weighing how selectively each neuron responds to the concept (Na et al., 2019). Selectivity is computed by taking a difference between average activation value of a neuron over a set of sentences where the underlying concept occurs and where it doesn't. A high selectivity value is obtained when a neuron is sensitive to the underlying concept and not to other concepts.

4.5 Qualitative Evaluation

Visualization has been used as a qualitative measure to evaluate the selected neurons. For example, Dalvi et al. (2019) visualized the top neurons and showed that they focus on very specific linguistic properties. They also visualized top-*k* activating words for the top neurons per concept to demonstrate the efficacy of their method. Visualization can be a very effective tool to evaluate the interpretations when it works in tandem with other methods—for example, using Concept Search or Probing-based methods to reduce the search space towards only highly activating concepts or the most salient neurons for these concepts, respectively.

5 Findings

Work done on neuron interpretation in NLP is predominantly focused on questions such as: *i) what concepts are learned within neurons? ii) how the knowledge is structured within representations?* We now iterate through various findings the above-described neuron analysis methods unravelled. Based on our main driving questions, we classify these into two broad categories: *i) concept discovery* and *ii) architectural analysis*.

5.1 Concept Discovery

In the following, we survey what lexical concepts or core-linguistic phenomenon are learned by the neurons in the network.

5.1.1 Lexical Concepts

Some of the research done on neuron analysis, particularly the work using visualization and concept search methods, identified neurons that capture lexical concepts.

Visualization Karpathy et al. (2015) found neurons that learn position of a word in the input sentence: Activating positively in the beginning, then becoming neutral in the middle and negatively towards the end. Li et al. (2016a) found intensification neurons that activate for words that intensify a sentiment. For example, ''I like this movie **a lot**'' or ''the movie is **incredibly** good''. Similarly, they discovered neurons that captured ''negation''. Both intensification neurons and sentiment neurons are relevant for the sentiment classification task, for which the understudied model was trained.

Concept Search Kádár et al. (2017) identified neurons that capture related groups of concepts in a multi-modal image captioning task. For example, they discovered neurons that learn electronic items ''camera, laptop, cables'' and salad items ''broccoli, noodles, carrots, etc''. Similarly, Na et al. (2019) found neurons that learn lexical concepts related to legislative terms (''law, legal, etc.). They also found neurons that learn phrasal concepts. Poerner et al. (2018) showed that *Concept Search* can be enhanced via *Corpus Generation*. They provided finer interpretation of the neurons by generating synthetic instances. For example, they showed that a ''horse racing'' neuron identified via concept search method was in fact a general ''racing'' neuron by generating novel contexts against this neuron.

5.1.2 Linguistic Concepts

A number of studies probed for neurons that capture core-linguistic concepts such as morphology, semantic tags, and so forth. Probing for linguistic structure is important to understand models' capacity to generalize (Marasović, 2018).[4] For example, the holy grail in machine translation is that a proficient model needs to be aware of word morphology, grammatical structure, and semantics to do well (Vauquois, 1968; Jones et al., 2012). Below we discuss major findings along this line of work.

[4]But is not the only reason to carry such an analysis.

Neurons specialize in core linguistic concepts. Dalvi et al. (2019), in their analysis of LSTM-based NMT models, found neurons that capture core linguistic concepts such as nouns, verb forms, numbers, articles, and so on. They also showed that **the number of neurons responsible for a concept varies based on the nature of the concept.** For example: closed class[5] concepts such as *Articles* (morphological category), *Months of Year* (semantic category) are localized to fewer neurons, whereas open class concepts such as *nouns* (morphological category) or *event* (semantic category) are distributed among a large number of neurons.

Neurons exhibit monosemous and polysemous behavior. Xin et al. (2019) found neurons exhibiting a variety of roles where a few neurons were exclusive to a single concept while others were polysemous in nature and captured several concepts. Suau et al. (2020) discovered neurons that capture different senses of a word. Similarly, Bau et al. (2019) found a switch neuron that activates positively for present-tense verbs and negatively for the past-tense verbs.

Neurons capture syntactic concepts and complex semantic concepts. Lakretz et al. (2019) discovered neurons that capture subject-verb agreement within LSTM gates. Karpathy et al. (2015) also found neurons that activate within quotes and brackets capturing long-range dependency. Na et al. (2019) aligned neurons with syntactic parses to show that neurons learn syntactic phrases. Seyffarth et al. (2021) analyzed complex semantic properties underlying a given sentence.

5.1.3 Salient Neurons for Models

In contrast to analyzing neurons with respect to a predefined concept, researchers also interpreted the concepts captured in the most salient neurons of the network. For example, in the analysis of the encoder of LSTM-based models, Bau et al. (2019) used Pearson correlation to discover salient neurons in the network. They found neurons that learn position of a word in the sentence

[5]Closed class concepts are part of language where new words are not added as the language evolves, for example functional words such as *can, be*, etc. In contrast, open class concepts are a pool where new words are constantly added as the language evolve, for example, ''chillax'' a verb formed blending ''chill'' and ''relax''.

among the most important neurons. Other neurons found included parentheses, punctuation, and conjunction neurons. Moreover, Li et al. (2016b) found that the two most salient neurons in *GloVe* were the frequency neurons that play an important role in all predictions.

The question of whether core-linguistic concepts are important for the end performance has been a less explored area. Dalvi et al. (2019) compared neurons learning morphological concepts and semantic concepts with unsupervised ranking of neurons with respect to their effect on the end performance. They found that the **model is more sensitive to the top neurons obtained using unsupervised ranking compared to linguistic concepts**. They showed that the unsupervised ranking of neurons is dominated by position information and other closed class categories such as conjunction and punctuation which according to the ablation experiment are more critical concepts for the end performance than linguistic concepts.

5.2 Architectural Analysis

Alongside studying what concepts are captured within deep NLP models, researchers have also studied: i) how these concepts are organized in the network? ii) how distributed and redundant they are? and iii) how this compares across architectures? Such an analysis is helpful in better understanding of the network and can be potentially useful in architectural search and model distillation.

5.2.1 Information Distribution

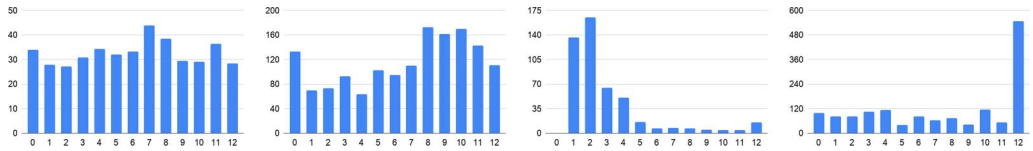
Human languages are hierarchical in structure where morphology and phonology sit at the bottom followed by lexemes, followed by syntactic structures. Concepts such as semantics and pragmatics are placed at the top of the hierarchy. Durrani et al. (2020) analyzed linguistic hierarchy by studying the spread of neurons across layers in various pretrained language models. They extracted salient neurons with respect to different linguistic concepts (e.g., morphology and syntax) and found that **neurons that capture word morphology were predominantly found in the lower and middle layers and those learning about syntax were found at the higher layers.** The observation was found to be true in both LSTM- and transformer-based architectures, and are in line with the findings of representation analysis (Liu et al., 2019; Tenney et al., 2019; Belinkov et al., 2020b). Similarly, Suau et al. (2020) analyzed sub-modules within GPT and RoBERTa transformer blocks and showed that lower layers within a transformer block accumulate more salient neurons than higher layers on the tasks of word sense disambiguation or homograph detection. They also found that the neurons that learn homographs are distributed across the network, as opposed to sense neurons that were more predominantly found at the lower layers.

5.2.2 Distributivity and Redundancy

While it is exciting to see that networks somewhat preserve linguistic hierarchy, many authors found that information is not discretely preserved at any individual layer, but is distributed and is redundantly present in the network. This is an artifact of various training choices such as dropout, which encourages the model to distribute knowledge across the network. For example, Li et al. (2016b) found specialized frequency neurons in a *GloVe* model trained without dropout, as opposed to the variant trained with dropout where the information was more redundantly available. Dalvi et al. (2020) showed that a significant amount of redundancy existed within pretrained models. They showed that 85% of the neurons across the network are redundant and at least 92% of the neurons can be removed when optimizing towards a downstream task in feature-based transfer learning.

5.2.3 Comparing Architectures

The distribution of neurons across the network has led researchers to draw interesting cross-architectural comparisons. Wu et al. (2020) performed correlation clustering of neurons across architectures and found that different architectures may have similar representations, but their individual neurons behave differently. Hennigen et al. (2020) compared neurons in contextualized (BERT) embedding with neurons in the static embedding (fastText) and found that fastText required two neurons to capture any morphosyntactic phenomenon, as opposed to BERT which required up to 35 neurons to obtain the same performance. Durrani et al. (2020) showed that the **linguistic knowledge in BERT (auto-encoder) is highly distributed across the network, as opposed to XLNet (auto-regressive) where neurons from a few layers are mainly responsible**



(a) POS – BERT (b) CCG – BERT (c) POS – XLNet (d) CCG – XLNet

Figure 2: Distribution of top neurons spread across different layers for each task. X-axis = Layer number, Y-axis = Number of neurons selected from that layer. Figure borrowed from Durrani et al. (2020).

for a concept (see Figure 2). Similarly, Suau et al. (2020) compared RoBERTa and GPT (auto-encoder vs. generative) models and found differences in the distribution of expert neurons. Durrani et al. (2021) extended the cross-architectural comparison towards fine-tuned models. They showed that after fine-tuning on GLUE tasks, the neurons capturing linguistic knowledge are regressed to lower layers in RoBERTa and XLNet, as opposed to BERT where it is still retained at the higher layers.

5.3 Summary of Findings

Below is a summary of the key findings that emerged from the work we covered in this survey. Neurons learned within Deep NLP models capture non-trivial linguistic knowledge ranging from lexical phenomenon such as morphemes, words, and multi-word expressions to highly complex global phenomenon such as semantic roles and syntactic dependencies. Neuron analysis resonates with the findings of representation analysis (Belinkov et al., 2017a,b; Tenney et al., 2019; Liu et al., 2019) in demonstrating that the networks follow linguistic hierarchy. Linguistic neurons are distributed across the network based on their complexity, with lower layers focused on the lexical concepts and middle and higher layers learning global phenomenon based on long-range contextual dependencies. While the networks preserve linguistic hierarchy, many authors showed that information is not discretely preserved, but is rather distributed and redundantly present in the network. It was also shown that a small optimal subset of neurons with respect to any concept can be extracted from a network. On another dimension, a few works showed that some concepts are localized to fewer neurons while others are distributed to a large group. Finally, some interesting cross architectural analyses were drawn based on how the neurons are distributed within their layers.

6 Applications

Neuron analysis leads to various applications beyond interpretation of deep models. In this section, we present several applications of neuron analysis: i) controlling model's behavior, ii) model distillation and efficiency, iii) domain adaptation, and iv) generating compositional explanations.

6.1 Controlling Model's Behavior

Once we have identified neurons that capture a certain concept learned in a model, these can be utilized for controlling the model's behavior with respect to that concept. Bau et al. (2019) identified *Switch Neurons* in NMT models that activate positively for the present-tense verbs and negatively for the past-tense verbs. By manipulating the values of these neurons, they were able to successfully change output translations from present to past tense during inference. The authors additionally found neurons that capture *gender* and *number agreement* concepts and manipulated them to control the system's output. Another effort along this line was carried out by Suau et al. (2020), where they manipulated the neurons responsible for a concept in the GPT model and generated sentences around specific topics of interest. Recently, Dai et al. (2021) manipulated salient neurons of relational facts and demonstrated their ability to update and erase knowledge about a particular fact. Controlling a model's behavior using neurons enables on-the-fly manipulation of output, for example, it can be used to debias the output of the model against sensitive attributes like race and gender.

6.2 Model Distillation and Efficiency

Deep NLP models are trained using hundreds of millions of parameters, limiting their applicability in computationally constrained environments. Identifying salient neurons and sub-networks can

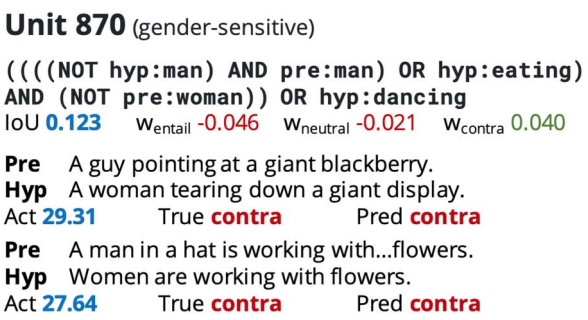


Figure 3: Compositional explanation using neuron 870 on the NLI task. Figure borrowed from Mu and Andreas (2020).

be useful for model distillation and efficiency. Dalvi et al. (2020) devised an efficient feature-based transfer learning procedure, stemming from their redundancy analysis. By exploiting layer and neuron-specific redundancy in the transformer models, they were able to reduce the feature set size to less than 10% neurons for several tasks while maintaining more than 97% of the performance. The procedure achieved a speedup of up to 6.2x in computation time for sequence labeling tasks as opposed to using all the features.

6.3 Domain Adaptation

Identifying the salient neurons with respect to a domain can be effectively used for domain adaptation and generalization. Gu et al. (2021) proposed a domain adaptation method using neuron pruning to target the problem of catastrophic forgetting of the general domain when fine-tuning a model for a target domain. They introduced a three-step adaptation process: i) rank neurons based on their importance, ii) prune the unimportant neurons from the network and retrain with student-teacher framework, and iii) expand the network to its original size and fine-tune towards in-domain, freezing the salient neurons and adjusting only the unimportant neurons. Using this approach helps to avoid catastrophic forgetting of the general domain while also obtaining optimal performance on the in-domain data.

6.4 Compositional Explanations

Knowing the association of a neuron with a concept enables explanation of a model's output. Mu and Andreas (2020) identified neurons that learn certain concepts in vision and NLP models. Using a composition of logical operators, they provided an explanation of model's prediction. Figure 3 presents an explanation using a gender-sensitive neuron. The neuron activates for contradiction when the premise contains the word *man*. Such explanations provide a way to generate adversarial examples that change model's predictions.

7 Open Issues and Future Directions

In the following section, we discuss several open issues and limitations related to methods, evaluation, and datasets. Moreover, we provide potential future directions vital to the progress of neuron and model interpretation.

- DNNs are distributed in nature, which encourages groups of neurons to work together to learn a concept. The current analysis methods, at large, ignore interaction between neurons while discovering neurons with respect to a concept. Trying all possible combination of neurons is a computationally intractable problem. A linear classifier using ElasticNet regularization (Dalvi et al., 2019) considers grouping of features during training—however, it's effectiveness in handling grouped neurons has not been empirically validated. Evolutionary algorithms[6] do not make any assumption of the underline distribution of the features and they have been effectively used for feature selection of multivariate features. Exploring them for neuron selection is a promising research direction to probe towards latent concepts in these models.

- A large number of interpretation studies rely on human-defined linguistic concepts to probe a model. It is possible that the models do not strictly adhere to the human-defined concepts and learn novel concepts about the language. This results in an incorrect or incomplete analysis. Several researchers (Michael et al., 2020; Dalvi et al., 2022; Sajjad et al., 2022) have made strides in this direction by analyzing hidden structures in the input representations in an unsupervised manner. They discovered the existence of novel structures not captured in the

[6] https://en.wikipedia.org/wiki/Evolutionary_algorithm.

human-defined categories. Dalvi et al. (2022) also proposed BERT ConceptNet, a manual annotation of the latent concepts in BERT. Introducing similar datasets across other models enables model-centric interpretation, and is a promising research direction.

- Although considerable work has been done on analyzing how knowledge is encoded within the learned representations, the question whether it is used by the model during prediction is a less explored area (Feder et al., 2021; Elazar et al., 2021). Ablation and knowledge attribution methods are two neuron interpretation methods that intrinsically use causal relation to select concept neurons. A few other studies evaluated the causal relation of the selected concept neurons via ablation or by clamping their activation values (Bau et al., 2019; Suau et al., 2020) and observed the change in model's prediction. However, most of the studies do not take into account the causal relation as part of the method or the evaluation of their method. The causal relation with respect to concept neurons is important to understand their importance to overall prediction and it leads way towards practical applications such as debiasing, model distillation, and domain adaptation.

- The work on neuron interpretation lacks standard evaluation benchmarks, and therefore studies conducted on identical models are not comparable. For example, there exists no gold annotation of neurons with respect to a certain dataset or a class. The curation of standard evaluation benchmarks is an essential step towards improving methods of interpretation of deep neural network models.

- The neuron analysis methods vary in their theoretical foundations as well as the perspective they aim to capture with respect to a given concept. This results in a selection of neurons that may not strictly align across all methods. For example, *Visualization*, *Neuron Search*, and *Corpus Search* discover neurons that are highly focused on a specific task (like ''less'' suffix or POS ''TO'' concepts), while *Probing-based* methods discover ranking of neurons that highlight grouping behavior within the neurons targeting broad concepts like POS ''Nouns''. Therefore, the choice of which neuron interpretation method to use is not straightforward and depends on various factors such as the nature of the concept to investigate, the availability of supervised data for the concept of interest etc. Apart from these high-level guiding principles, a thorough comparison of methods with respect to the nature of the concept of interest is needed to fully understand the strengths and weaknesses of each approach. Antverg and Belinkov (2022) is one such effort in this direction that compares three neuron interpretation methods.

- Neuron-level interpretation opens the door for a number of applications useful for the successful deployment of DNN systems (Section 6). However, most of the research conducted in this direction is preliminary. For example, there are many open research questions in **controlling a system's behavior** using neurons such as: i) are all concepts manipulatable? ii) how to identify neurons that can be controlled to change the output? iii) is high distributiveness a hindrance for controlling model's behavior? and iv) whether disentangled (Bengio et al., 2012) and sparse models (Frankle and Carbin, 2019) may serve as a better alternate on this front? Addressing these questions will enable a more reliable control of the deep NLP models and entails numerous applications such as removing bias and adapting the system to novel domains.

References

Yossi Adi, Einat Kermany, Yonatan Belinkov, Ofer Lavi, and Yoav Goldberg. 2016. Fine-grained analysis of sentence embeddings using auxiliary prediction tasks. *arXiv preprint arXiv:1608.04207*.

J. Alammar. 2020. Interfaces for explaining transformer language models.

J. Alammar. 2021. Ecco: An open source library for the explainability of transformer language models. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing: System Demonstrations*, pages 249–257,

Online. Association for Computational Linguistics. https://doi.org/10.18653/v1/2021.acl-demo.30

Rana Ali Amjad, Kairen Liu, and Bernhard C. Geiger. 2018. Understanding neural networks and individual neuron importance via information-ordered cumulative ablation.

Omer Antverg and Yonatan Belinkov. 2022. On the pitfalls of analyzing individual neurons in language models. In *International Conference on Learning Representations*.

Omer Antverg, Eyal Ben-David, and Yonatan Belinkov. 2022. Idani: Inference-time domain adaptation via neuron-level interventions. https://doi.org/10.18653/v1/2022.deeplo-1.3

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*.

Anthony Bau, Yonatan Belinkov, Hassan Sajjad, Nadir Durrani, Fahim Dalvi, and James Glass. 2019. Identifying and controlling important neurons in neural machine translation. In *International Conference on Learning Representations*.

Yonatan Belinkov, Nadir Durrani, Fahim Dalvi, Hassan Sajjad, and James Glass. 2017a. What do neural machine translation models learn about morphology? In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (ACL)*, Vancouver. Association for Computational Linguistics. https://doi.org/10.18653/v1/P17-1080

Yonatan Belinkov, Nadir Durrani, Fahim Dalvi, Hassan Sajjad, and James Glass. 2020a. On the linguistic representational power of neural machine translation models. *Computational Linguistics*, 45(1):1–57. https://doi.org/10.1162/coli_a_00367

Yonatan Belinkov, Nadir Durrani, Fahim Dalvi, Hassan Sajjad, and James Glass. 2020b. On the linguistic representational power of neural machine translation models. *Computational Linguistics*, 46(1):1–52. https://doi.org/10.1162/coli_a_00367

Yonatan Belinkov, Lluís Màrquez, Hassan Sajjad, Nadir Durrani, Fahim Dalvi, and James Glass. 2017b. Evaluating layers of representation in neural machine translation on part-of-speech and semantic tagging tasks. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1–10, Taipei, Taiwan. Asian Federation of Natural Language Processing.

Yoshua Bengio, Aaron C. Courville, and Pascal Vincent. 2012. Unsupervised feature learning and deep learning: A review and new perspectives. *CoRR*, abs/1206.5538.

Maxim Binshtok, Ronen I. Brafman, Solomon Eyal Shimony, Ajay Martin, and Crag Boutilier. 2007. Computing optimal subsets. In *Proceedings of the Twenty Second AAAI Conference on Artificial Intelligence (AAAI, Oral presentation)*.

Alexis Conneau, German Kruszewski, Guillaume Lample, Loïc Barrault, and Marco Baroni. 2018. What you can cram into a single vector: Probing sentence embeddings for linguistic properties. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (ACL)*. https://doi.org/10.18653/v1/P18-1198

Damai Dai, Li Dong, Yaru Hao, Zhifang Sui, and Furu Wei. 2021. Knowledge neurons in pretrained transformers. *CoRR*, abs/2104.08696.

Fahim Dalvi, Nadir Durrani, Hassan Sajjad, Yonatan Belinkov, D. 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 Thirty-Third AAAI Conference on Artificial Intelligence (AAAI, Oral presentation)*. https://doi.org/10.1609/aaai.v33i01.33016309

Fahim Dalvi, Abdul Rafae Khan, Firoj Alam, Nadir Durrani, Jia Xu, and Hassan Sajjad. 2022. Discovering latent concepts learned in BERT. In *International Conference on Learning Representations*.

Fahim Dalvi, Hassan Sajjad, Nadir Durrani, and Yonatan Belinkov. 2020. Analyzing redundancy in pretrained transformer models. In

Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP-2020), pages 4908–4926, Online.

Nicola De Cao, Michael Sejr Schlichtkrull, Wilker Aziz, and Ivan Titov. 2020. How do decisions emerge across layers in neural models? Interpretation with differentiable masking. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 3243–3255, Online. Association for Computational Linguistics. https://doi.org/10.18653/v1/2020.emnlp-main.262

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 (Long and Short Papers)*, Minneapolis, Minnesota. Association for Computational Linguistics.

Kedar Dhamdhere, Ashish Agarwal, and Mukund Sundararajan. 2020. The shapley taylor interaction index.

Kedar Dhamdhere, Mukund Sundararajan, and Qiqi Yan. 2018. How important is a neuron? *CoRR*, abs/1805.12233.

Nadir Durrani, Fahim Dalvi, and Hassan Sajjad. 2022. Linguistic correlation analysis: Discovering salient neurons in deep NLP models.

Nadir Durrani, Hassan Sajjad, and Fahim Dalvi. 2021. How transfer learning impacts linguistic knowledge in deep NLP models? In *Findings of the Association for Computational Linguistics: ACL 2021*, pages 4947–4957, Online. Association for Computational Linguistics. https://doi.org/10.18653/v1/2021.findings-acl.438

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 (EMNLP)*, pages 4865–4880, Online. Association for Computational Linguistics. https://doi.org/10.18653/v1/2020.emnlp-main.395

Yanai Elazar, Shauli Ravfogel, Alon Jacovi, and Yoav Goldberg. 2021. Amnesic probing: Behavioral explanation with amnesic counterfactuals. *Transactions of the Association for Computational Linguistics*, 9:160–175. https://doi.org/10.1162/tacl_a_00359

Dumitru Erhan, Yoshua Bengio, Aaron Courville, and Pascal Vincent. 2009. Visualizing higher-layer features of a deep network. Technical Report 1341, University of Montreal. Also presented at the ICML 2009 Workshop on Learning Feature Hierarchies, Montreal, Canada.

Manaal Faruqui, Yulia Tsvetkov, Dani Yogatama, Chris Dyer, and Noah A. Smith. 2015. Sparse overcomplete word vector representations. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1491–1500, Beijing, China. Association for Computational Linguistics. https://doi.org/10.3115/v1/P15-1144

Amir Feder, Nadav Oved, Uri Shalit, and Roi Reichart. 2021. CausaLM: Causal model explanation through counterfactual language models. *Computational Linguistics*, 47(2):333–386. https://doi.org/10.1162/coli_a_00404

Shi Feng, Eric Wallace, Alvin Grissom II, Mohit Iyyer, Pedro Rodriguez, and Jordan Boyd-Graber. 2018. Pathologies of neural models make interpretations difficult. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3719–3728, Brussels, Belgium. Association for Computational Linguistics. https://doi.org/10.18653/v1/D18-1407

Jonathan Frankle and Michael Carbin. 2019. The lottery ticket hypothesis: Finding sparse, trainable neural networks. In *International Conference on Learning Representations*.

Alona Fyshe, Leila Wehbe, Partha P. Talukdar, Brian Murphy, and Tom M. Mitchell. 2015. A compositional and interpretable semantic space. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 32–41, Denver,

Colorado. Association for Computational Linguistics. https://doi.org/10.3115/v1/N15-1004

Fréderic Godin, Kris Demuynck, Joni Dambre, Wesley De Neve, and Thomas Demeester. 2018. Explaining character-aware neural networks for word-level prediction: Do they discover linguistic rules? In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3275–3284, Brussels, Belgium. Association for Computational Linguistics. https://doi.org/10.18653/v1/D18-1365

Shuhao Gu, Yang Feng, and Wanying Xie. 2021. Pruning-then-expanding model for domain adaptation of neural machine translation.

Lucas Torroba Hennigen, Adina Williams, and Ryan Cotterell. 2020. Intrinsic probing through dimension selection. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 197–216, Online. Association for Computational Linguistics. https://doi.org/10.18653/v1/2020.emnlp-main.15

John Hewitt and Percy Liang. 2019. Designing and interpreting probes with control tasks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2733–2743, Hong Kong, China. https://doi.org/10.18653/v1/D19-1275

Dieuwke Hupkes. 2020. Hierarchy and interpretability in neural models of language processing. Ph.D. thesis. University of Amsterdam.

Dieuwke Hupkes, Sara Veldhoen, and Willem Zuidema. 2018. Visualisation and 'diagnostic classifiers' reveal how recurrent and recursive neural networks process hierarchical structure.

Bevan Jones, Jacob Andreas, Daniel Bauer, Karl Moritz Hermann, and Kevin Knight. 2012. Semantics-based machine translation with hyperedge replacement grammars. In *Proceedings of COLING 2012*, pages 1359–1376, Mumbai, India. The COLING 2012 Organizing Committee.

Ákos Kádár, Grzegorz Chrupała, and Afra Alishahi. 2017. Representation of linguistic form and function in recurrent neural networks. *Computational Linguistics*, 43(4):761–780. https://doi.org/10.1162/COLI_a_00300

Andrej Karpathy, Justin Johnson, and Li Fei-Fei. 2015. Visualizing and understanding recurrent networks. *arXiv preprint arXiv:1506.02078*.

Yair Lakretz, German Kruszewski, Theo 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 (Long and Short Papers)*, pages 11–20, Minneapolis, Minnesota. Association for Computational Linguistics. https://doi.org/10.18653/v1/N19-1002

Jiwei Li, Xinlei Chen, Eduard Hovy, and Dan Jurafsky. 2016a. Visualizing and understanding neural models in NLP. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 681–691, San Diego, California. Association for Computational Linguistics.

Jiwei Li, Will Monroe, and Dan Jurafsky. 2016b. Understanding neural networks through representation erasure. *CoRR*, abs/1612.08220.

Zachary C. Lipton. 2016. The mythos of model interpretability. In *ICML Workshop on Human Interpretability in Machine Learning (WHI)*.

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 (Long and Short Papers)*, pages 1073–1094, Minneapolis, Minnesota. Association for Computational Linguistics.

Scott M. Lundberg and Su-In Lee. 2017. A unified approach to interpreting model predictions. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R.

Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 4765–4774. Curran Associates, Inc.

Ana Marasović. 2018. NLP's generalization problem, and how researchers are tackling it. *The Gradient*.

Leland Mclnnes, John Healy, and James Melville. 2020. UMAP: Uniform manifold approximation and projection for dimension reduction.

Richard Meyes, Constantin Waubert de Puiseau, Andres Posada-Moreno, and Tobias Meisen. 2020. Under the hood of neural networks: Characterizing learned representations by functional neuron populations and network ablations. *CoRR*, abs/2004.01254.

Julian Michael, Jan A. Botha, and Ian Tenney. 2020. Asking without telling: Exploring latent ontologies in contextual representations. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6792–6812, Online. Association for Computational Linguistics. https://doi.org/10.18653/v1/2020.emnlp-main.552

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. In *Proceedings of the ICLR Workshop*, Scottsdale, AZ, USA.

Jesse Mu and Jacob Andreas. 2020. Compositional explanations of neurons. *CoRR*, abs/2006.14032.

W. James Murdoch, Peter J. Liu, and Bin Yu. 2018. Beyond word importance: Contextual decomposition to extract interactions from lstms.

Seil Na, Yo Joong Choe, Dong-Hyun Lee, and Gunhee Kim. 2019. Discovery of natural language concepts in individual units of CNNs. *CoRR*, abs/1902.07249.

Chris Olah, Arvind Satyanarayan, Ian Johnson, Shan Carter, Ludwig Schubert, Katherine Ye, and Alexander Mordvintsev. 2018. The building blocks of interpretability. *Distill*. https://distill.pub/2018/building-blocks. https://doi.org/10.23915/distill.00010

Tiago Pimentel, Josef Valvoda, Rowan Hall Maudslay, Ran Zmigrod, Adina Williams, and Ryan Cotterell. 2020. Information-theoretic probing for linguistic structure. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4609–4622, Online. Association for Computational Linguistics. https://doi.org/10.18653/v1/2020.acl-main.420

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, Brussels, Belgium. Association for Computational Linguistics. https://doi.org/10.18653/v1/W18-5437

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners. *OpenAI Blog*, 1(8):9.

Maithra Raghu, Justin Gilmer, Jason Yosinski, and Jascha Sohl-Dickstein. 2017, SVCCA: Singular vector canonical correlation analysis for deep learning dynamics and interpretability. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 6078–6087. Curran Associates, Inc.

Hassan Sajjad, Firoj Alam, Fahim Dalvi, and Nadir Durrani. 2021. Effect of post-processing on contextualized word representations. *CoRR*, abs/2104.07456.

Hassan Sajjad, Nadir Durrani, Fahim Dalvi, Firoj Alam, Abdul Khan, and Jia Xu. 2022. Analyzing encoded concepts in transformer language models. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3082–3101, Seattle, United States. Association for Computational Linguistics. https://doi.org/10.18653/v1/2022.naacl-main.225

Esther Seyffarth, Younes Samih, Laura Kallmeyer, and Hassan Sajjad. 2021. Implicit representations of event properties within contextual language models: Searching for

''causativity neurons''. In *International Conference on Computational Semantics (IWCS)*.

Karolina Stanczak, Lucas Torroba Hennigen, Adina Williams, Ryan Cotterell, and Isabelle Augenstein. 2022. A latent-variable model for intrinsic probing. *CoRR*, abs/2201.08214.

Wolfgang G. Stock. 2010. Concepts and semantic relations in information science. *Journal of the American Society for Information Science and Technology*, 61(10):1951–1969. https://doi.org/10.1002/asi.21382

Xavier Suau, Luca Zappella, and Nicholas Apostoloff. 2020. Finding experts in transformer models. *CoRR*, abs/2005.07647.

Mukund Sundararajan, Ankur Taly, and Qiqi Yan. 2017. Axiomatic attribution for deep networks.

Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. 2014. Sequence to sequence learning with neural networks. *CoRR*, abs/1409.3215.

Ian Tenney, Dipanjan Das, and Ellie Pavlick. 2019. BERT rediscovers the classical NLP pipeline. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4593–4601, Florence, Italy. Association for Computational Linguistics. https://doi.org/10.18653/v1/P19-1452

Ke Tran, Arianna Bisazza, and Christof Monz. 2018. The importance of being recurrent for modeling hierarchical structure. *arXiv preprint arXiv:1803.03585*. https://doi.org/10.18653/v1/D18-1503

Mehrdad Valipour, En-Shiun Annie Lee, Jaime R. Jamacaro, and Carolina Bessega. 2019. Unsupervised transfer learning via BERT neuron selection. *CoRR*, abs/1912.05308.

Bernard Vauquois. 1968. A survey of formal grammars and algorithms for recognition and transformation in mechanical translation. In *IFIP Congress (2)*, pages 1114–1122.

Elena Voita and Ivan Titov. 2020. Information-theoretic probing with minimum description length. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

John Wu, Yonatan Belinkov, Hassan Sajjad, Nadir Durrani, Fahim Dalvi, and James Glass. 2020. Similarity analysis of contextual word representation models. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL)*, Seattle. Association for Computational Linguistics.

Ji Xin, Jimmy Lin, and Yaoliang Yu. 2019. What part of the neural network does this? Understanding LSTMs by measuring and dissecting neurons. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5823–5830, Hong Kong, China. Association for Computational Linguistics. https://doi.org/10.18653/v1/D19-1591

Kelly Zhang and Samuel Bowman. 2018. Language modeling teaches you more than translation does: Lessons learned through auxiliary syntactic task analysis. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 359–361, Brussels, Belgium. Association for Computational Linguistics. https://doi.org/10.18653/v1/W18-5448

Luisa M. Zintgraf, Taco S. Cohen, Tameem Adel, and Max Welling. 2017. Visualizing deep neural network decisions: Prediction difference analysis. *CoRR*, abs/1702.04595.

	Scope	Input	Output	Scalability	HITL	Supervision	Causation
Visualization							
Karpathy et al. (2015) Fyshe et al. (2015); Faruqui et al. (2015); Li et al. (2016a)	local	neuron	concept	low	yes	no	no
Corpus-based methods							
Concept Search							
Kádár et al. (2017)	global	neuron	concept	low	yes	no	no
Na et al. (2019)	global	neuron	concept	high	no	no	no
Neuron Search							
Mu and Andreas (2020); Suau et al. (2020); Antverg and Belinkov (2022); Antverg et al. (2022)	global	concept	neurons	high	no	yes	no
Probing-based methods							
Linear (Radford et al., 2019; Dalvi et al., 2019; Lakretz et al., 2019; Durrani et al., 2020, 2021, 2022; Hupkes, 2020; Antverg et al., 2022)	global	concept	neurons	high	no	yes	no
Random Forest (Valipour et al., 2019)	global	concept	neurons	high	no	yes	no
Gaussian (Hennigen et al., 2020; Stanczak et al., 2022)	global	concept	neurons	high	no	yes	no
Causation-based methods							
Ablation (Li et al., 2016a; Amjad et al., 2018; Xin et al., 2019; Lakretz et al., 2019)	both	concept/ class	neurons	medium	no	no	yes
Knowledge attribution (Dhamdhere et al., 2018, 2020; Lundberg and Lee, 2017; Tran et al., 2018; Dai et al., 2021; Murdoch et al., 2018; Godin et al., 2018)	local	concept/ class	neurons	high	no	no	yes
Miscellaneous methods							
Corpus generation (Poerner et al., 2018)	global	neuron	concept	low	yes	no	no
Matrix factorization (Alammar, 2020, 2021)	local	neurons	neurons	low	yes	no	no
Clustering (Dalvi et al., 2020)	global	neurons	neurons	high	yes	no	no
Multi model search (Bau et al., 2019; Wu et al., 2020)	global	neurons	neurons	high	yes	no	no

Table 3: Comparison of neuron analysis methods based on various attributes. The exhaustive list of citations for each method are provided in the text.