# From Neural Activations to Concepts: A Survey on Explaining Concepts in Neural Networks

Jae Hee Lee *, Sergio Lanza and Stefan Wermter
*Knowledge Technology Group, Department of Informatics, University of Hamburg, Germany*
*E-mails: jae.hee.lee@uni-hamburg.de, sergio.lanza@uni-hamburg.de, stefan.wermter@uni-hamburg.de*

**Abstract.** In this paper, we review recent approaches for explaining *concepts* in neural networks. Concepts can act as a natural link between learning and reasoning: once the concepts are identified that a neural learning system uses, one can integrate those concepts with a reasoning system for inference or use a reasoning system to act upon them to improve or enhance the learning system. On the other hand, knowledge can not only be extracted from neural networks but concept knowledge can also be inserted into neural network architectures. Since integrating learning and reasoning is at the core of neuro-symbolic AI, the insights gained from this survey can serve as an important step towards realizing neuro-symbolic AI based on explainable concepts.

Keywords: Explainable artificial intelligence, concept explanation, neuro-symbolic integration

## 1. Introduction

In recent years, neural networks have been successful in tasks that were regarded to require human-level intelligence, such as understanding and generating images and texts, performing dialogues, and controlling robots to follow instruction [1–3]. However, their decision-making is often *not* explainable, which undermines user trust and negatively impacts their usage in sensitive or critical domains, such as automation, law, and medicine. One way to overcome this limitation is by making neural networks explainable, e.g., by designing them to generate explanations or by using a *post-hoc* explanation method that analyzes the behavior of a neural network after it has been trained.

This paper reviews explainable artificial intelligence (XAI) methods with a focus on explaining how neural networks learn *concepts*, as concepts can act as primitives for building complex rules, presenting themselves as a natural link between learning and reasoning [4], which is at the core of neuro-symbolic AI [5–9]. On the one hand, identifying the concepts that a neural network uses for a given input can inform the user about what information the network is using to generate its output [10–15]. Combined with an approach to extract all relevant concepts and their (causal) relationships, one could generate explanations in logical or natural language that faithfully reflects the decision procedure of the network. On the other hand, the identified concepts can help a symbolic reasoner intervene in the neural network such that debugging the network becomes possible by modification of the concepts [11, 16–18].

Some XAI surveys have been published in recent years [19–25]. However, almost all of them are mainly concerned with the use of saliency maps to highlight important input features. Only a few surveys include concept explanation as a way to explain neural networks. A recent survey in this vein is by Casper et al. [26], which discusses a broad

*Corresponding author. E-mail: jae.hee.lee@uni-hamburg.de.

range of approaches to explaining the internals of neural networks. However, due to its broader scope, the survey does not provide detailed descriptions of methods for explaining concepts and misses recent advances in the field. The surveys by Schwalbe [27] and Sajjad et al. [28], on the other hand, are dedicated to specific kinds of concept explanation methods with a focus on either vision [27] or natural language processing [28] and are, therefore, limited in scope, failing to analyze the two areas together.

We categorize concept explanation approaches and structure this survey based on whether they explain concepts at the level of individual neurons (Section 2) or at the level of layers (Section 3). The last section summarizes this survey with open questions.

## 2. Neuron-Level Explanations

The smallest entity in a neural network that can represent a concept is a *neuron* [28], which could be—in a broader sense—also a *unit* or a *filter* in a convolutional neural network [10]. In this section, we survey approaches that explain, in a post-hoc manner, concepts that a neuron of a pre-trained neural network represents, either by comparing the *similarity* between a concept and the activation of the neuron (see Section 2.1) or by detecting the *causal relationship* between a concept and the activation of the neuron (see Section 2.2).

### *2.1. Using Similarities between Concepts and Activations*

In this category, the concept a neuron is representing is explained by comparing the concept with the activations of the neuron when the concept is passed as an input to the model. The *network dissection* approach by Bau et al. [10] is arguably the most prominent approach in this category, which is mainly applied to computer vision models. In this approach, a set $\mathcal{C}$ of concepts are prepared as well as a set $\mathcal{X}_C$ of images for each concept $C \in \mathcal{C}$. Then the activations of a convolutional filter are measured for each input $x \in \mathcal{X}_C$. Afterward, the activation map is thresholded to generate a binary activation mask $M(x)$ and scaled up to be compared with the original concept (e.g., concept `head`) in the binary segmentation mask $L_C(x)$ of the input $X$ (e.g., the `head` segment of an image with a bird). See Figure 1 for an illustration. Then, to measure to which degree concept $C$ is represented by the convolutional filter, the dataset-wide intersection over union metric (IoU) is computed, which is defined as $\text{IoU}(C) = \sum_{x\in\mathcal{X}_C} |M(x) \cap L_C(x)| / \sum_{x\in\mathcal{X}_C} |M(x) \cup L_C(x)|$. If the IoU value is above a given threshold, then the convolutional filter represents the concept $C$. Several extensions of this approach have been introduced. Fong et al. [29] question whether a concept has to be represented by a single convolutional filter alone or whether it can be represented by a linear combination of filters. They show that the latter leads to a better representation of the concept and also suggest to use binary classification for measuring how well filters represent a concept. Complementary to that extension, Mu et al. [13] investigate how to approximate better what a single filter represents. To this end, they assume that a filter can represent a boolean combination of concepts (e.g., `(water OR river) AND NOT blue`) and show that this compositional explanation of concepts leads to higher IoU. An intuitive extension of using compositional explanations is using *natural language* explanations. The approach called MILAN by Hernandez et al. [14] finds such natural language explanations as a sequence $d$ of words that maximizes the pointwise mutual information between $d$ and a set of image regions $E$ that maximally activates the filter, (i.e., $\arg\max_d \log P(d \mid E) - \log P(d)$). In the approach, the two probabilities $P(d \mid E)$ and $\log P(d)$ are approximated by an image captioning model and a language model, respectively, which are trained on a dataset that the authors curated.

One strong assumption made by the network dissection approach is the availability of a comprehensive set $\mathcal{C}$ of concepts and corresponding labeled images to provide accurate explanations of neurons. This is, however, difficult to obtain in general. Oikarinen et al. [30] tackle this problem with their CLIP-Dissect method, which is based on the CLIP [31] vision-language model. (CLIP embeds images and texts in the same vector space, allowing for measuring the similarity between texts and images.) To explain the concept a convolutional filter $k$ is representing, they choose a set $\mathcal{X}_k$ of the most highly activating images for filter $k$, then use CLIP to measure the similarity between $\mathcal{X}_k$ and each concept $C \in \mathcal{C}$ (here, the concept set $\mathcal{C}$ consists of 20K most common English words), and finally find the best matching concept $C$.

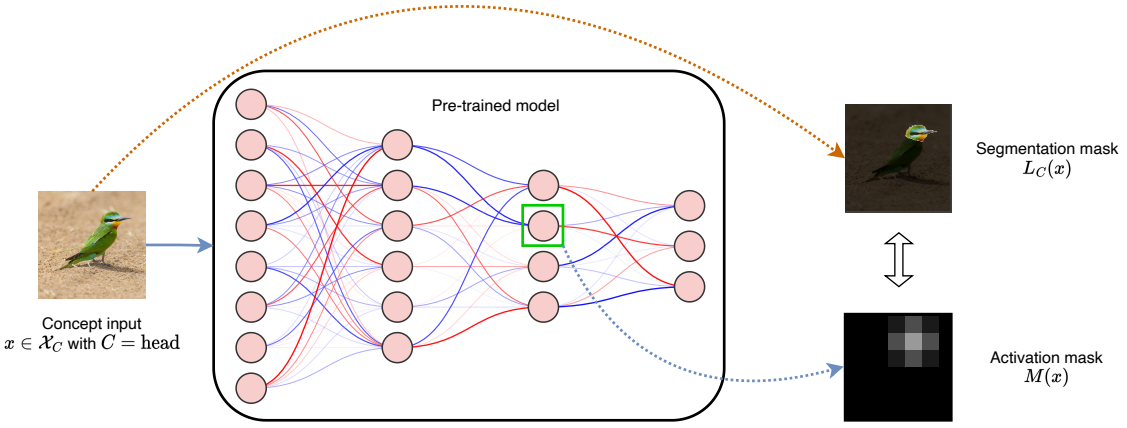


Fig. 1. Neuron-level explanation using similarities between concepts and activations. Depicted is the network dissection approach, which compares the segmented concept in the input with the activation mask of a neuron [10].

The dissection approach can also be used in generative vision models. Bau et al. [16] identify that units of generative adversarial networks [32] learn concepts similar to network dissection and that one can intervene on the units and remove specific concepts to change the output image (e.g., removing units representing the concept `tree` leads to output images with fewer trees in their scenes).

### 2.2. *Using Causal Relationships between Concepts and Activations*

In this category, the concepts that a neuron is representing are explained by analyzing the causal relationship either (i) between the input concept and the neuron by intervening on the input and measuring the neural activation or (ii) between the neuron and the output concept by intervening on the neural activation and measuring the probability in predicting the concept. This approach is often used for explaining neurons of NLP models [28], where the types of concepts can be broader (e.g., subject-verb behavior, causal relationship, semantic tags).

The first line of work investigates the influence of a concept in the input on the activation of a neuron by intervening in the input. Kádár et al. [33] find the $n$-grams (i.e., a sequence of $n$ words) that have the largest influence on the activation of a neuron by measuring the change in its activations when a word is removed from the $n$-grams. Na et al. [34] first identify $k$ sentences that most highly activate a filter of a CNN-based NLP model. From these $k$ sentences, they extract concepts by breaking down each sentence into a set of consecutive word sequences that form a meaningful chunk. Then they measure the contribution of each concept to the filters' activations by first repeating the concept to create a synthetic sentence of a fixed length (to normalize the input's contribution to the unit across different concepts) and then measuring the mean value of the filter's activations.

The second line of work investigates the role of a neuron in generating a concept by intervening in the activation of the neuron. Dai et al. [35] investigate the factual linguistic knowledge of the BERT model [36], a widely used pre-trained model for text classification, which is pre-trained among other tasks by predicting masked words in a sentence. In this approach, given relational facts with a mask word (e.g. "Rome is the capital of [MASK]"), each neuron's contribution to predicting the mask is measured using the integrated gradients method [37]. To verify the causal role of the neuron that is supposed to represent a concept, the authors also intervene in the neuron's activation (by suppressing or doubling) and measure the change in accuracy in predicting the concept. Finlayson et al. [38] analyze whether a neuron of a transformer-based language model (e.g., GPT-2 [39]) has acquired the concept of conjugation. The authors determine which neuron contributes most to the conjugation of a verb by using the causal mediation analysis [40]. To this end, they first modify the activation of a neuron to the one that the neuron would have output if there was an intervention on the input (e.g., the subject in the input sentence was changed from singular to plural) and then measure the amount of change between the predictions of the correct conjugation of a verb with and without the intervention (see Figure 2). Meng et al. [18] also apply causal mediation analysis to GPT-2 to understand which neurons memorize factual knowledge and modify specific facts (e.g., "The Eiffel Tower is in Paris" is modified

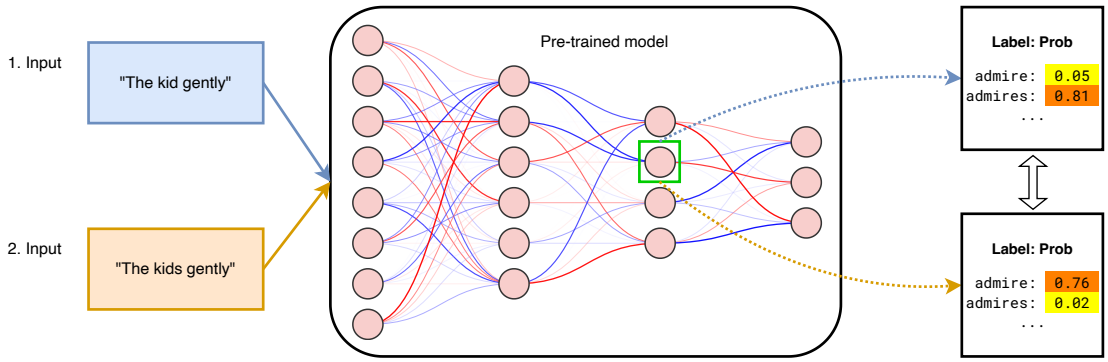


Fig. 2. Neuron-level explanation using causal relationships between concepts and activations. In causal mediation analysis, the activation of a neuron is modified to the one that the neuron would have output if there was an intervention on the input (the subject in the input sentence was changed from singular to plural). After, it is measured the amount of change between the predictions of the correct conjugation of a verb with and without the intervention [38].

to "The Eiffel Tower is in Rome"). The data they use consists of triples of the form (*subject*, *relation*, *object*) and the model has to predict the *object* given *subject* and *relation*. They discover that the neurons in the middle layer feed-forward modules in GPT-2 are the most relevant for encoding factual information and implementing a weight modifier to change the value of weights and alter the factual knowledge.

## 3. Layer-Level Explanations

Concepts can also be represented by a whole *layer* as opposed to a neuron or a convolutional filter, as mentioned to in the paragraph about the work by Fong et al. [29] in Section 2.1. This can be achieved in a post-hoc manner for a pre-trained model by passing examples of a concept dataset $\mathcal{C}$ to the model and extracting the activations of a specific layer to train a concept classifier. Two approaches are prominent in layer-level explanations: the first is explaining with *concept activation vectors* (CAVs) (see Section 3.1) and the second is *probing* (see Section 3.2). The main difference between the two approaches is that in the case of CAV a linear binary classifier is trained for each concept $C \in \mathcal{C}$, and in probing a multiclass classifier is trained with classification labels that are often related to certain linguistic features (e.g, sentiments, part-of-speech tags). On the other hand, concepts can be baked in a layer, where each concept represents a neuron as was done with localist representations in the early days of neural network research (see Section 3.3).

### *3.1. Using Vectors to Explain Concepts: Concept Activation Vectors*

A *concept activation vector* (CAV) introduced by Kim et al. [12] is a continuous vector that corresponds to a concept represented by a layer of a neural network $f$ (see Figure 3). Let $f = f^{\top} \circ f^{\perp}$, where $f^{\perp} : \mathbb{R}^m \to \mathbb{R}^n$ is the bottom part of the network whose final convolutional layer $\ell$ is of interest. To identify the existence of a concept $C$ (e.g., the concept `stripes`) in layer $\ell$, network $f^{\perp}$ is first fed with positive examples $x_C^+$ that contain concept $C$ and negative examples $x_C^-$ that do not contain the concept, and then their corresponding activations $f^{\perp}(x_C^+) \in \mathbb{R}^n$ and $f^{\perp}(x_C^-) \in \mathbb{R}^n$ are collected. Next, a linear classifier is learned that distinguishes activations $f^{\perp}(x_C^+)$ from activations $f^{\perp}(x_C^-)$. The vector normal $v_C \in \mathbb{R}^n$ to the decision boundary of the classifier is then a CAV of concept $C$. One useful feature of a CAV is that it allows for testing how much an input image $x$ is correlated with a concept $C$ (e.g., an image of a zebra and concept `stripes`), which is called *testing with CAVs* (TCAV) in [12]. This is accomplished, roughly speaking, by measuring the probability of a concept $C$ having a positive influence on predicting a class label $k \in \{1, \ldots, K\}$ on a dataset $\mathcal{X}$, i.e., how much moving the latent vector $f^{\perp}(x) \in \mathbb{R}^n$ along the direction of $v_C$, i.e., $f^{\perp}(x) + \epsilon \cdot v_C$, changes the log-probability of label $k$ when it is fed to $f^{\top}$ for all images $x \in \mathcal{X}$ with class label $k$.

CAVs can be used in many different ways. Nejagholi et al. [41] use CAVs to identify sensitivity of abusive language classifiers wrt. implicit types (as opposed to explicit types) of abusive language. Different from the original

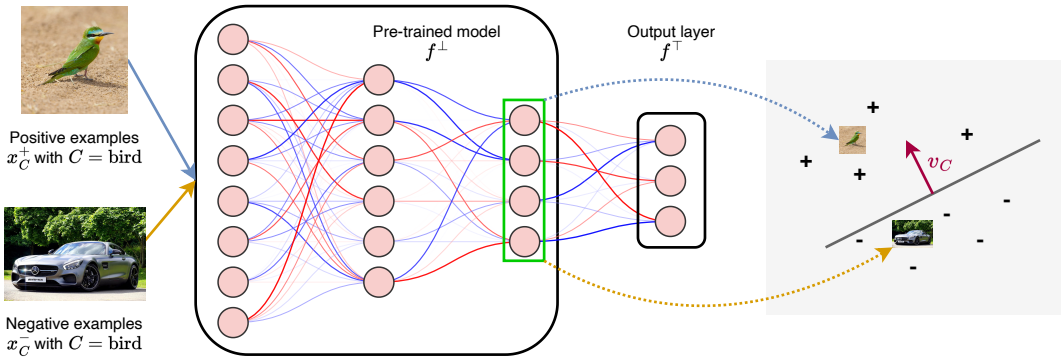


Fig. 3. Layer-level explanation using vectors to explain concepts. For each concept $C$ positive examples $x_C^+$ and negative examples $x_C^-$ are fed to a pre-trained model to learn the so-called *concept activation vector* (CAV) $v_C$ from the corresponding activations of the target layer [12].

approach [12] which obtains CAVs by taking the vector normal to the decision boundary, they obtain CAVs by just averaging over the activations $f^\perp(x_C^+)$ for all positive samples $x_C^+$ to mitigate the impact of the choice of random negative samples $x_C^-$ on determining the decision boundary. Zhou et al. [42] decompose the row vector of the last linear layer for predicting a class label $k$ and represent it as a linear combination of a basis that consists of CAVs using only positive weights. Each positive weight then indicates how much of the corresponding concept is involved in predicting class label $k$. Similarly, Abid et al. [17] propose an approach that learns a set of CAVs, but for debugging purposes. Given an input image misclassified by a model, a weighted sum of the set of CAVs is computed that leads to correct classification when added to the activations before the last linear layer of the model. In addition to explaining bugs on a conceptual level, this approach allows for identifying spurious correlations in the data.

An issue with the original approach for learning CAVs is that one needs to prepare a set of concept labels and images to learn the CAVs. Ghorbani et al. [43] partially tackle this issue by preparing images of the same class and then segmenting them with multiple resolutions. The clusters of resulting segments then form concepts and can be used for TCAV. As corresponding concept labels are missing, the concepts need to be manually inspected. Yeh et al. [44] circumvent the problem of preparing a concept dataset by training CAVs together with a model on the original image classification dataset. To this end, they compute a vector-valued score, where each value corresponds to a learnable concept and indicates to which degree the concept is present in the receptive field of the convolutional layer (computed by building a scalar product). The score is then passed to a multilayer perceptron (MLP) to perform classification.

### *3.2. Using Classifiers to Explain Concepts: Probing*

Similar to the CAV-based approaches in Section 3.1, *probing* uses a classifier to explain concepts. However, instead of training a binary linear classifier for each concept $C \in \mathcal{C}$ to measure the existence of the concept in the activation of a layer, probing uses a classifier for multiclass classifications with labels that often represent linguistic features in NLP (e.g., sentiments, part-of-speech tags). For example, given sentences as inputs to a pre-trained NLP model (e.g., BERT [36]), probing allows for evaluating how well the sentence embeddings of the model capture certain syntactic and semantic information, such as the length or the tense of the sentence [45–47] (see Figure 3.3).

Probing, which is designed as a layer-level explanation method, can also be combined with neuron-level explanation method (see Section 2) by applying the probing classifier only to neurons that are relevant for the classification [48]. Finding such neurons can be accomplished by applying the elastic-net regularization to the classifier, which constrains both the L1 and the L2-norm of the classifier weights.

The concepts learned by such probing classifiers can be combined with a knowledge base to provide richer explanations. Ribiero and Leite [49] use identified concepts as evidence to draw conclusions from a set of axioms in a knowledge base (e.g., given an axiom $\texttt{LongFreightTrain} \leftarrow \texttt{LongTrain} \wedge \texttt{FreightTrain}$ in the knowledge base, identifying both antecedent concepts `LongTrain` and `FreightTrain` in the activations explains the presence of the consequence `LongFreightTrain` in the input). However, one cannot always assume the

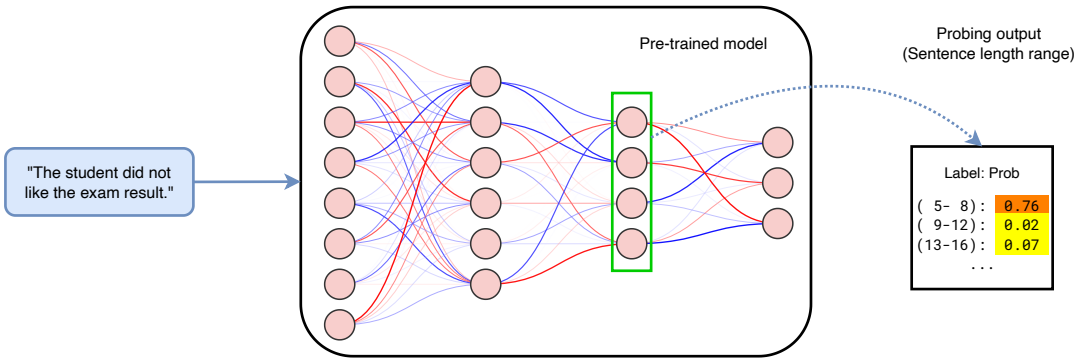


Fig. 4. Layer-level explanation using a classifier to explain concepts. In this example, a pre-trained model takes as its input a sentence and a probing classifier is applied to the activation of the highlighted layer to check whether the activation encodes the concept of sentence length [46].

presence of a knowledge base for a given task. Ferreira et al. [50] weaken this assumption by learning the underlying theory from the identified concept using an induction framework.

Since the probing classifier is trained independently from the pre-trained model, it was pointed out that the pre-trained model does not necessarily leverage the same features that the classifier uses for predicting a given concept, i.e., what the probing classifier detects can be merely a correlation between the activation and the concept [51, 52].

### *3.3. Using Localist Representations: Concept Bottleneck Models*

Different from the neuron-based approach in Section 2, where concepts are learned in a post-hoc manner, in a *concept bottleneck model* (CBM) [11], each concept is represented by a unique neuron in the bottleneck layer $f^\ell$ of a model $f$ (see Figure 5), which is a reminiscence of localist representation [53]. This layer provides information about the existence or the strengths of each concept in the input. The output of the bottleneck layer is then used by a classifier or regressor $f^\top$ for the prediction, which allows for explaining what concept led to the given prediction. Often, the bottom part $f^\perp$ of a pre-trained model is used for initializing the layers before the concept bottleneck $f^\ell$ and $f^\ell$ is a linear layer that maps the features from $f^\perp$ to concepts. Therefore, a CBM is $f = f^\top \circ f^\ell \circ f^\perp$. To train the concept bottleneck $f^\ell$, the training data has to include concept labels in addition to task labels.

One of the main limitations of CBMs is the need for the aforementioned concept labels, which might not be available for specific tasks. Several recent approaches overcome this limitation [54–56]. The main idea behind these approaches is using an external resource to obtain a set $\mathcal{C}$ of concepts relevant to the task. This external resource could be a knowledge base such as ConceptNet [54], or the 20K common words English words [55], or a language model like GPT-3 [56]. After obtaining concept set $\mathcal{C}$, each concept word $C \in \mathcal{C}$ is embedded to a vector $v_C$ by means of the CLIP vision-language model (cf. Section 2.1) such that vector $v_C$ can be used for computing the strength of concept $C$ for a given input $x \in \mathcal{X}$, e.g., by measuring the cosine similarity between $v_C$ and the embedding $f^\perp(x)$. Finally, the presence of concepts in the concept bottleneck layers allows for inducing logical explanations, e.g., Ciravegna et al. [57] induce explanations in disjunctive normal form (DNF) from concept activations and predicted labels, which is similar to logic-based explanation approaches [49, 50] in Section 3.2.

## 4. Conclusion

In this survey, we have reviewed recent methods for explaining concepts in neural networks. We have covered different approaches that range from analyzing individual neurons to learning classifiers for a whole layer. As witnessed by the increasing number of recent papers, this is an active research area and a lot is still to be discovered, for example, empirically comparing or integrating different approaches[1]. With the progress of concept extraction

[1]For example, concept bottleneck models in Section 3.3 and concept activation vectors in Section 3.1 have been combined by Yuksekgonul et al. [54].

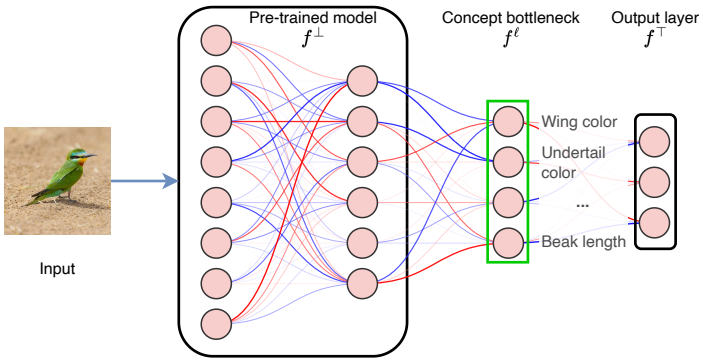


Fig. 5. Layer-level explanation using the concept bottleneck model approach [11]. Each neuron in the concept bottleneck $f^{\ell}$ corresponds to a unique concept (e.g., `wing color`).

from neural networks, integrating the learned neural concepts with symbolic representations—also known as *neuro-symbolic integration*—is receiving (again) increasing attention [49, 50, 57, 58]. In conclusion, this line of research is still very active and in development, providing ample opportunities for new forms of integration in neuro-symbolic AI.

## 5. Acknowledgement

The authors gratefully acknowledge support from the DFG (CML, MoReSpace, LeCAREbot), BMWK (SIDIMO, VERIKAS), and the European Commission (TRAIL, TERAIS). We would like to thank Cornelius Weber for valuable comments on this paper

## References

[1] OpenAI, GPT-4 Technical Report, *arXiv preprint arXiv:2303.08774* (2023). http://arxiv.org/abs/2303.08774.
[2] A. Ramesh, M. Pavlov, G. Goh, S. Gray, C. Voss, A. Radford, M. Chen and I. Sutskever, Zero-Shot Text-to-Image Generation, *arXiv:2102.12092 [cs]* (2021). http://arxiv.org/abs/2102.12092.
[3] Y. Jiang, A. Gupta, Z. Zhang, G. Wang, Y. Dou, Y. Chen, L. Fei-Fei, A. Anandkumar, Y. Zhu and L. Fan, VIMA: Robot Manipulation with Multimodal Prompts, in: *Proceedings of the 40th International Conference on Machine Learning*, PMLR, 2023, pp. 14975–15022. ISSN 2640-3498. https://proceedings.mlr.press/v202/jiang23b.html.
[4] S. Schockaert and V. Gutiérrez-Basulto, Modelling Symbolic Knowledge Using Neural Representations, in: *Reasoning Web. Declarative Artificial Intelligence : 17th International Summer School 2021, Leuven, Belgium, September 8–15, 2021, Tutorial Lectures*, M. Šimkus and I. Varzinczak, eds, Lecture Notes in Computer Science, Springer International Publishing, Cham, 2022, pp. 59–75. ISBN 978-3-030-95481-9. https://doi.org/10.1007/978-3-030-95481-9_3.
[5] P. Hitzler, M.K. Sarker and A. Eberhart (eds), *Compendium of Neurosymbolic Artificial Intelligence*, Frontiers in Artificial Intelligence and Applications / Faia Vol. 369, IOS Press, Washington, 2023. ISBN 978-1-64368-406-2.
[6] S. Wermter and W.G. Lehnert, A Hybrid Symbolic/Connectionist Model for Noun Phrase Understanding, *Connection Science* **1**(3) (1989), 255–272. https://doi.org/10.1080/09540098908915641.
[7] S. Wermter, C. Panchev and G. Arevian, Hybrid Neural Plausibility Networks for News Agents, in: *Proceedings of the National Conference on Artificial Intelligence AAAI*, 1999, pp. 93–98. https://cdn.aaai.org/AAAI/1999/AAAI99-014.pdf.
[8] K.J. McGarry, J. Tait, S. Wermter and J. MacIntyre, Rule-Extraction from Radial Basis Function Networks, in: *International Conference on Artificial Neural Networks*, Vol. 2, 1999, pp. 613–618. ISSN 0537-9989.
[9] J.H. Lee, M. Sioutis, K. Ahrens, M. Alirezaie, M. Kerzel and S. Wermter, Chapter 19. Neuro-Symbolic Spatio-Temporal Reasoning, in: *Compendium of Neurosymbolic Artificial Intelligence*, IOS Press, 2023, pp. 410–429. https://ebooks.iospress.nl/doi/10.3233/FAIA230151.
[10] D. Bau, B. Zhou, A. Khosla, A. Oliva and A. Torralba, Network Dissection: Quantifying Interpretability of Deep Visual Representations, in: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 6541–6549. https://openaccess.thecvf.com/content_cvpr_2017/html/Bau_Network_Dissection_Quantifying_CVPR_2017_paper.html.
[11] P.W. Koh, T. Nguyen, Y.S. Tang, S. Mussmann, E. Pierson, B. Kim and P. Liang, Concept Bottleneck Models, in: *Proceedings of the 37th International Conference on Machine Learning*, PMLR, 2020, pp. 5338–5348. ISSN 2640-3498. https://proceedings.mlr.press/v119/koh20a.html.

[12] B. Kim, M. Wattenberg, J. Gilmer, C. Cai, J. Wexler, F. Viegas and R. Sayres, Interpretability Beyond Feature Attribution: Quantitative Testing with Concept Activation Vectors (TCAV), in: *Proceedings of the 35th International Conference on Machine Learning*, PMLR, 2018, pp. 2668–2677. ISSN 2640-3498. https://proceedings.mlr.press/v80/kim18d.html.

[13] J. Mu and J. Andreas, Compositional Explanations of Neurons, in: *Advances in Neural Information Processing Systems*, Vol. 33, Curran Associates, Inc., 2020, pp. 17153–17163. https://proceedings.neurips.cc/paper/2020/hash/c74956ffb38ba48ed6ce977af6727275-Abstract.html.

[14] E. Hernandez, S. Schwettmann, D. Bau, T. Bagashvili, A. Torralba and J. Andreas, Natural Language Descriptions of Deep Visual Features, in: *International Conference on Learning Representations*, 2022. https://openreview.net/forum?id=NudBMY-tzDr.

[15] S. Wermter, Knowledge Extraction from Transducer Neural Networks, *Applied Intelligence* **12**(1) (2000), 27–42. https://doi.org/10.1023/A:1008320219610.

[16] D. Bau, J.-Y. Zhu, H. Strobelt, B. Zhou, J.B. Tenenbaum, W.T. Freeman and A. Torralba, GAN Dissection: Visualizing and Understanding Generative Adversarial Networks, in: *International Conference on Learning Representations*, 2018. https://openreview.net/forum?id=Hyg_X2C5FX.

[17] A. Abid, M. Yuksekgonul and J. Zou, Meaningfully Debugging Model Mistakes Using Conceptual Counterfactual Explanations, in: *Proceedings of the 39th International Conference on Machine Learning*, PMLR, 2022, pp. 66–88. ISSN 2640-3498. https://proceedings.mlr.press/v162/abid22a.html.

[18] K. Meng, D. Bau, A. Andonian and Y. Belinkov, Locating and Editing Factual Associations in GPT, *Advances in Neural Information Processing Systems* **35** (2022), 17359–17372. https://proceedings.neurips.cc/paper_files/paper/2022/hash/6f1d43d5a82a37e89b0665b33bf3a182-Abstract-Conference.html.

[19] R. Dwivedi, D. Dave, H. Naik, S. Singhal, R. Omer, P. Patel, B. Qian, Z. Wen, T. Shah, G. Morgan and R. Ranjan, Explainable AI (XAI): Core Ideas, Techniques, and Solutions, *ACM Computing Surveys* **55**(9) (2023), 194:1–194:33. https://doi.org/10.1145/3561048.

[20] R. Ibrahim and M.O. Shafiq, Explainable Convolutional Neural Networks: A Taxonomy, Review, and Future Directions, *ACM Computing Surveys* **55**(10) (2023), 206:1–206:37. https://dl.acm.org/doi/10.1145/3563691.

[21] A. Madsen, S. Reddy and S. Chandar, Post-Hoc Interpretability for Neural NLP: A Survey, *ACM Computing Surveys* **55**(8) (2022), 155:1–155:42. https://dl.acm.org/doi/10.1145/3546577.

[22] G. Ras, N. Xie, M. van Gerven and D. Doran, Explainable Deep Learning: A Field Guide for the Uninitiated, *Journal of Artificial Intelligence Research* **73** (2022), 329–396. https://www.jair.org/index.php/jair/article/view/13200.

[23] X. Li, H. Xiong, X. Li, X. Wu, X. Zhang, J. Liu, J. Bian and D. Dou, Interpretable Deep Learning: Interpretation, Interpretability, Trustworthiness, and Beyond, *Knowledge and Information Systems* **64**(12) (2022), 3197–3234. https://doi.org/10.1007/s10115-022-01756-8.

[24] W. Samek, G. Montavon, S. Lapuschkin, C.J. Anders and K.-R. Müller, Explaining Deep Neural Networks and Beyond: A Review of Methods and Applications, *Proceedings of the IEEE* **109**(3) (2021), 247–278.

[25] F. Sado, C.K. Loo, W.S. Liew, M. Kerzel and S. Wermter, Explainable Goal-driven Agents and Robots - A Comprehensive Review, *ACM Computing Surveys* **55**(10) (2023), 211:1–211:41. https://doi.org/10.1145/3564240.

[26] S. Casper, T. Rauker, A. Ho and D. Hadfield-Menell, Toward Transparent AI: A Survey on Interpreting the Inner Structures of Deep Neural Networks, in: *First IEEE Conference on Secure and Trustworthy Machine Learning*, 2023. https://openreview.net/forum?id=8C5zt-0Utdn.

[27] G. Schwalbe, Concept Embedding Analysis: A Review, *arXiv preprint arXiv:2203.13909* (2022). http://arxiv.org/abs/2203.13909.

[28] H. Sajjad, N. Durrani and F. Dalvi, Neuron-Level Interpretation of Deep NLP Models: A Survey, *Transactions of the Association for Computational Linguistics* **10** (2022), 1285–1303. https://doi.org/10.1162/tacl_a_00519.

[29] R. Fong and A. Vedaldi, Net2Vec: Quantifying and Explaining How Concepts Are Encoded by Filters in Deep Neural Networks, in: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 8730–8738. https://openaccess.thecvf.com/content_cvpr_2018/html/Fong_Net2Vec_Quantifying_and_CVPR_2018_paper.html.

[30] T. Oikarinen and T.-W. Weng, CLIP-Dissect: Automatic Description of Neuron Representations in Deep Vision Networks, in: *The Eleventh International Conference on Learning Representations*, 2023. https://openreview.net/forum?id=iPWiwWHc1V.

[31] A. Radford, J.W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, G. Krueger and I. Sutskever, Learning Transferable Visual Models From Natural Language Supervision, in: *Proceedings of the 38th International Conference on Machine Learning*, PMLR, 2021, pp. 8748–8763. ISSN 2640-3498. https://proceedings.mlr.press/v139/radford21a.html.

[32] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville and Y. Bengio, Generative Adversarial Nets, in: *Advances in Neural Information Processing Systems 27*, Z. Ghahramani, M. Welling, C. Cortes, N.D. Lawrence and K.Q. Weinberger, eds, Curran Associates, Inc., 2014, pp. 2672–2680. http://papers.nips.cc/paper/5423-generative-adversarial-nets.pdf.

[33] Á. Kádár, G. Chrupała and A. Alishahi, Representation of Linguistic Form and Function in Recurrent Neural Networks, *Computational Linguistics* **43**(4) (2017), 761–780. https://doi.org/10.1162/COLI_a_00300.

[34] S. Na, Y.J. Choe, D.-H. Lee and G. Kim, Discovery of Natural Language Concepts in Individual Units of CNNs, in: *International Conference on Learning Representations*, 2018. https://openreview.net/forum?id=S1EERs09YQ.

[35] D. Dai, L. Dong, Y. Hao, Z. Sui, B. Chang and F. Wei, Knowledge Neurons in Pretrained Transformers, in: *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Association for Computational Linguistics, Dublin, Ireland, 2022, pp. 8493–8502. https://aclanthology.org/2022.acl-long.581.

[36] J. Devlin, M.-W. Chang, K. Lee and K. Toutanova, 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)*, Association for Computational Linguistics, Minneapolis, Minnesota, 2019, pp. 4171–4186. https://aclanthology.org/N19-1423.

[37] M. Sundararajan, A. Taly and Q. Yan, Axiomatic Attribution for Deep Networks, in: *Proceedings of the 34th International Conference on Machine Learning*, PMLR, 2017, pp. 3319–3328. ISSN 2640-3498. https://proceedings.mlr.press/v70/sundararajan17a.html.

[38] M. Finlayson, A. Mueller, S. Gehrmann, S. Shieber, T. Linzen and Y. Belinkov, Causal Analysis of Syntactic Agreement Mechanisms in Neural 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 (Volume 1: Long Papers)*, Association for Computational Linguistics, Online, 2021, pp. 1828–1843. https://aclanthology.org/2021.acl-long.144.

[39] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei and I. Sutskever, Language Models Are Unsupervised Multitask Learners, 2019. https://cdn.openai.com/better-language-models/language_models_are_unsupervised_multitask_learners.pdf.

[40] J. Vig, S. Gehrmann, Y. Belinkov, S. Qian, D. Nevo, Y. Singer and S. Shieber, Investigating Gender Bias in Language Models Using Causal Mediation Analysis, in: *Advances in Neural Information Processing Systems*, Vol. 33, Curran Associates, Inc., 2020, pp. 12388–12401. https://proceedings.neurips.cc/paper/2020/hash/92650b2e92217715fe312e6fa7b90d82-Abstract.html.

[41] I. Nejadgholi, K. Fraser and S. Kiritchenko, Improving Generalizability in Implicitly Abusive Language Detection with Concept Activation Vectors, in: *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Association for Computational Linguistics, Dublin, Ireland, 2022, pp. 5517–5529. https://aclanthology.org/2022.acl-long.378.

[42] B. Zhou, Y. Sun, D. Bau and A. Torralba, Interpretable Basis Decomposition for Visual Explanation, in: *Computer Vision – ECCV 2018*, V. Ferrari, M. Hebert, C. Sminchisescu and Y. Weiss, eds, Lecture Notes in Computer Science, Springer International Publishing, Cham, 2018, pp. 122–138. ISBN 978-3-030-01237-3.

[43] A. Ghorbani, J. Wexler, J.Y. Zou and B. Kim, Towards Automatic Concept-based Explanations, in: *Advances in Neural Information Processing Systems*, Vol. 32, Curran Associates, Inc., 2019. https://papers.nips.cc/paper_files/paper/2019/hash/77d2afcb31f6493e350fca61764efb9a-Abstract.html.

[44] C.-K. Yeh, B. Kim, S. Arik, C.-L. Li, T. Pfister and P. Ravikumar, On Completeness-aware Concept-Based Explanations in Deep Neural Networks, in: *Advances in Neural Information Processing Systems*, Vol. 33, Curran Associates, Inc., 2020, pp. 20554–20565. https://proceedings.neurips.cc/paper/2020/hash/ecb287ff763c169694f682af52c1f309-Abstract.html.

[45] A. Ettinger, A. Elgohary and P. Resnik, Probing for Semantic Evidence of Composition by Means of Simple Classification Tasks, in: *Proceedings of the 1st Workshop on Evaluating Vector-Space Representations for NLP*, Association for Computational Linguistics, Berlin, Germany, 2016, pp. 134–139. https://aclanthology.org/W16-2524.

[46] Y. Adi, E. Kermany, Y. Belinkov, O. Lavi and Y. Goldberg, Fine-Grained Analysis of Sentence Embeddings Using Auxiliary Prediction Tasks, in: *International Conference on Learning Representations*, 2016. https://openreview.net/forum?id=BJh6Ztuxl.

[47] A. Conneau, G. Kruszewski, G. Lample, L. Barrault and M. Baroni, What You Can Cram into a Single \backslash$&!#* Vector: Probing Sentence Embeddings for Linguistic Properties, in: *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Association for Computational Linguistics, Melbourne, Australia, 2018, pp. 2126–2136. http://aclweb.org/anthology/P18-1198.

[48] N. Durrani, H. Sajjad, F. Dalvi and Y. Belinkov, Analyzing Individual Neurons in Pre-trained Language Models, in: *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Association for Computational Linguistics, Online, 2020, pp. 4865–4880. https://aclanthology.org/2020.emnlp-main.395.

[49] M.d.S. Ribeiro and J. Leite, Aligning Artificial Neural Networks and Ontologies towards Explainable AI, *Proceedings of the AAAI Conference on Artificial Intelligence* **35**(6) (2021), 4932–4940. https://ojs.aaai.org/index.php/AAAI/article/view/16626.

[50] J. Ferreira, M.d.S. Ribeiro, R. Gonçalves and J. Leite, Looking Inside the Black-Box: Logic-based Explanations for Neural Networks, *Proceedings of the International Conference on Principles of Knowledge Representation and Reasoning* **19**(1) (2022), 432–442. https://proceedings.kr.org/2022/45/.

[51] Y. Belinkov, Probing Classifiers: Promises, Shortcomings, and Advances, *Computational Linguistics* **48**(1) (2022), 207–219. https://aclanthology.org/2022.cl-1.7.

[52] A. Amini, T. Pimentel, C. Meister and R. Cotterell, Naturalistic Causal Probing for Morpho-Syntax, *arXiv preprint arXiv:2205.07043* (2022). http://arxiv.org/abs/2205.07043.

[53] M. Page, Connectionist Modelling in Psychology: A Localist Manifesto, *Behavioral and Brain Sciences* **23**(4) (2000), 443–467. https://www.cambridge.org/core/journals/behavioral-and-brain-sciences/article/abs/connectionist-modelling-in-psychology-a-localist-manifesto/65F9E3CEC90E0C80A46B25E0028BCFE3.

[54] M. Yuksekgonul, M. Wang and J. Zou, Post-Hoc Concept Bottleneck Models, in: *The Eleventh International Conference on Learning Representations*, 2023. https://openreview.net/forum?id=nA5AZ8CEyow.

[55] T. Oikarinen, S. Das, L.M. Nguyen and T.-W. Weng, Label-Free Concept Bottleneck Models, in: *The Eleventh International Conference on Learning Representations*, 2023. https://openreview.net/forum?id=FlCg47MNvBA.

[56] Y. Yang, A. Panagopoulou, S. Zhou, D. Jin, C. Callison-Burch and M. Yatskar, Language in a Bottle: Language Model Guided Concept Bottlenecks for Interpretable Image Classification, in: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 19187–19197. https://openaccess.thecvf.com/content/CVPR2023/html/Yang_Language_in_a_Bottle_Language_Model_Guided_Concept_Bottlenecks_for_CVPR_2023_paper.html.

[57] G. Ciravegna, P. Barbiero, F. Giannini, M. Gori, P. Liò, M. Maggini and S. Melacci, Logic Explained Networks, *Artificial Intelligence* (2022), 103822. https://www.sciencedirect.com/science/article/pii/S000437022200162X.

[58] A. Dalal, M.K. Sarker, A. Barua, E. Vasserman and P. Hitzler, Understanding CNN Hidden Neuron Activations Using Structured Background Knowledge and Deductive Reasoning, *arXiv preprint arXiv:2308.03999* (2023). http://arxiv.org/abs/2308.03999.