Interactive Explanations of Internal Representations of Neural Network Layers: An Exploratory Study on Outcome Prediction of Comatose Patients

Meike Nauta[1] and **Michel J.A.M. van Putten**[1,2] and **Marleen C. Tjepkema-Cloostermans**[2] and **Jeroen Peter Bos**[1] and **Maurice van Keulen**[1] and **Christin Seifert**[1]

Abstract. Supervised machine learning models have impressive predictive capabilities, making them useful to support human decision-making. However, most advanced machine learning techniques, such as Artificial Neural Networks (ANNs), are black boxes and therefore not interpretable for humans. A way of explaining an ANN is visualizing the internal feature representations of its hidden layers (neural embeddings). However, interpreting these visualizations is still difficult. We therefore present InterVENE: an approach that visualizes neural embeddings and interactively explains this visualization, aiming for knowledge extraction and network interpretation. We project neural embeddings in a 2-dimensional scatter plot, where users can interactively select two subsets of data instances in this visualization. Subsequently, a personalized decision tree is trained to distinguish these two sets, thus explaining the difference between the two sets. We apply InterVENE to a medical case study where interpretability of decision support is critical: outcome prediction of comatose patients. Our experiments confirm that InterVENE can successfully extract knowledge from an ANN, and give both domain experts and machine learning experts insight into the behaviour of an ANN. Furthermore, InterVENE's explanations about outcome prediction of comatose patients seem plausible when compared to existing neurological domain knowledge.

1 Introduction

Most advanced artificial intelligence techniques, such as Artificial Neural Networks (ANNs), are black boxes and therefore not interpretable for humans. As argued by [22], it depends on the application to what extent this is a concern. Interpretability is critical in the case of this paper: outcome prediction of comatose patients, where AI is meant to support the physician in (high-stakes) decision making. "Explainable AI" (XAI) is essential for medical professionals to understand the how and why of the AI's decision, for example, to be able to confirm that the system is right for the right reasons [23]. Moreover, interpretable algorithms (i.e. algorithms that explain or present their decision to a human in understandable terms [5]) could appropriately enhance users' trust in future AI systems [22]. Since a poor predicted diagnosis may cause care to be reduced, getting insights in the algorithm may reveal predictions that cannot be trusted, thus allowing to save a patient's life. On the other hand, AI might discover patterns that were not known to the medical profession before, leading to knowledge discovery for healthcare improvement.

One way of explaining an ANN is visualizing the internal feature representations of its hidden layers (called *neural embeddings*) [26]. Solely relying on such visualization techniques is, however, insufficient: the resulting projection does not explicitly show the relations between projected points and the original input features, making it challenging to understand why data points are placed far apart or close together. Interviews revealed that data analysts try to map the synthetic data dimensions to original input features, and that they try to name and verify clusters [3]. In this paper, we therefore *explain a visualization*.

Contributions Since visualizations of neural embeddings are too complex to readily grasp, we explain a visualization with an easy-to-understand and effective interactive approach, suitable for both domain and machine learning experts. After visualizing neural embeddings in a scatter plot with dimensionality-reduction, a user can interact with the visualization by selecting two sets of data points in the visualization. We explain the difference between these subsets by training a decision tree (or another interpretable model) that distinguishes between the user-selected subsets in the visualization in terms of the original input features. Allowing manual selection of data points in the visualization results in a personalized explanation that gives meaning to clusters seen in the visualization where the user had specific interest in. An overview of the overall process is shown in Figure 1.

Our approach serves two goals: (i) knowledge discovery and knowledge validation: extracting knowledge from the neural network allows the domain expert to observe and validate patterns learnt by the neural network, and (ii) network interpretation: understanding the neural network allows the machine learning expert to analyse errors, behaviour over training time and contributions of single layers.

We implemented our approach in a tool called InterVENE (Interactively Visualizing and Explaining Neural Embeddings)[3]. We apply InterVENE to the medical domain where interpretability of a decision support system is critical: outcome prediction for postanoxic coma patients, based on a structured dataset containing features of EEG recordings of 518 comatose patients.

[1] University of Twente, the Netherlands, email addresses: {m.nauta, m.j.a.m.vanputten, m.vankeulen, c.seifert}@utwente.nl, j.p.bos@student.utwente.nl

[2] Department of Neurology and Clinical Neurophysiology, Medisch Spectrum Twente, the Netherlands, email addresses: {m.vanputten, m.tjepkema-cloostermans}@mst.nl

[3] InterVENE is open-sourced at `https://github.com/M-Nauta/InterVENE`

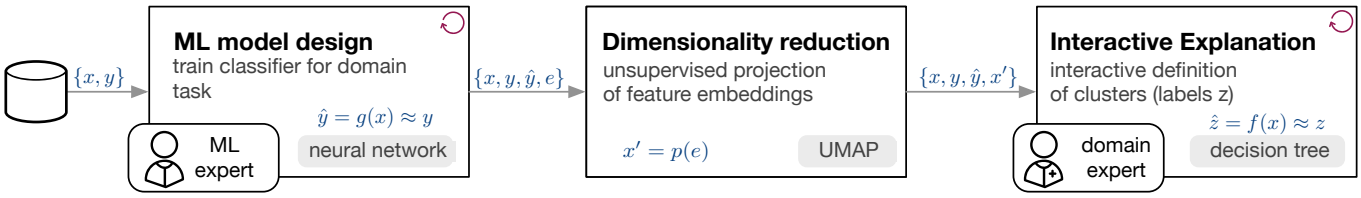


Figure 1. Overview of the approach. A machine learning model (e.g. an artificial neural network) is trained, the feature vectors of its hidden layers (embeddings) are projected into 2D. Domain experts then can interactively explore the visualizations and select groups of interest, whose difference is explained by an interpretable discriminative model (e.g. a decision tree).

2 Related Work

Explanation approaches of machine learning models address different stakeholders and explain different aspects of the model. Three general approaches have emerged towards providing explanations [7]. First, explanations can be model-based by showing the operational procedure of the complete model. Some machine learning models are considered inherently interpretable, e.g. decision trees or decision rules [6] or can be extended to be inherently interpretable, such as Deep Neural Decision Trees [31]. More complex models, such as deep neural networks can be locally or globally approximated by interpretable models (e.g., [8]). Other explanations approaches show similar cases (e.g., [20]), or the contributions of features for a decision (e.g., [12]). In this paper, we use decision trees as interpretable models to locally explain subgroups that emerge from the implicit feature representations of end-to-end machine learning models.

Dimensionality reduction techniques are suitable for visualizing high-dimensional data by projecting it on a low-dimensional space while preserving as much of the original data structure. Although many projection techniques have been proposed (we refer the reader to [14] for a review), t-SNE [15] is arguably the best known and most applied technique, due to its capability of capturing both the local and global structure of the high-dimensional data. It provides a 2- or 3-dimensional feature representation that can be visualized in a plot. Recently, the Uniform Manifold Approximation and Projection (UMAP) algorithm was introduced [16], which preserves as much of the local and more of the global data structure than t-SNE with faster run times [2]. UMAP uses the nearest neighbour descent algorithm [4] to construct a weighted k-neighbour graph that approximates the representation of the high dimensional data. It then optimizes this low dimensional layout via probabilistic edge sampling and negative sampling [17]. It has been shown that UMAP provides meaningful visualizations for biological data [2], materials science [13] and image classification [16].

Dimensionality reduction can also be applied to **derived features**, such as embeddings. An embedding is a vector containing the post-activation values in a hidden layer of an Artificial Neural Network (ANN). The learned, continuous embedding is therefore a representation of the input data. Visualizing the embeddings of each hidden layer provides a general overview of the inner behavior of the ANN [25]. Most existing work on visualizing embeddings was created for image data, using e.g. heat maps or pixel displays [25]. Dimension-reduction techniques such as t-SNE and UMAP however, can be used for visualizing the embeddings of any type of data. Data instances with similar representations in a network layer will be close in the projection space. It has already been shown that t-SNE projections of neural embeddings (both after training and during training) can aid the understanding and improving of ANNs [21], since it allows users to view global geometry and to discover clusters [26]. The embedding projector of [26], using t-SNE or Principal Component Analysis, is therefore integrated into the Tensorflow platform [1].

Although dimension-reduction is considered an *explainable* method because data is represented in a lower-dimensional space, users often have difficulty **interpreting the dimensions of the visualization** in a meaningful way [14]. It is therefore needed to *explain the visualization*. For example, [27] introduced *probing*, a tool to understand projections by exploring the dimensionality-reduced data and to interact with the visualization to examine errors. [11] explains a visualization by creating just-in-time descriptions to automatically identify and annotate visual features, such as clusters and outliers. In contrast, our approach provides personalized explanations by explaining the difference between user-chosen clusters with an interpretable model. Our approach can be applied to any dimensionality reduction method that visualizes embeddings, such as t-SNE [15], UMAP [16], DarkSight [30] or the existing embedding projector of [27].

Reliable **prediction of neurological outcome in comatose patients** after cardiac arrest is challenging. It may prevent futile care in patients with a poor prognosis, and allow timely communication with family members about the neurological condition. Early EEG recordings have been shown to allow reliable prognostication within 24 hours after arrest in a significant fraction of patients [9]. However, visual analysis by the neurologist is time-consuming and allows reliable prediction of neurological outcome of approximately 50% of patients, only. This motivates the need for techniques that assist or even replace the human expert, as resources are limited, and hold promise to increase the diagnostic yield. Deep neural networks have recently been used to predict neurological outcome [29], showing a significant improvement in classification accuracy. A limitation of these approaches, however, is the lack of interpretability, which is what we address in this paper.

3 Approach

Our approach consists of three general steps, integrating two stakeholders: the machine learning expert and the domain expert. As shown in Figure 1, the machine learning expert first **trains a predictive model** for the task at hand on the labeled data $\{x, y\}$, with x being the feature vector and y the respective label. This task is iterative and usually includes model selection and hyper-parameter optimisation steps. The output is the machine learning model, and its predictions $\hat{y}$ on the data, which ideally should match the ground-truth labels y. For the remainder of this paper, we assume that these models generate an implicit feature representation to solve their prediction task. In end-to-end (deep) learning scenarios these feature representations are the activations in the hidden layers of the neural networks. We refer to these representations as embeddings e. There can be more than one embedding for one training sample, e.g. the embeddings of the first hidden layer, of the second and so on. Em-

beddings can also be collected for different training epochs, e.g. after training the network for 10 epochs versus training for 50 epochs.

The embeddings of a layer of a certain epoch are then **projected into a 2-dimensional feature space**, resulting in one visualization for each layer. More specifically, the projection function p takes embeddings e from feature vector x and generates a low-dimensional representation x'. In principle, any projection of dimensionality reduction method can be used in this step. We choose a 2-dimensional scatter plot as visualization since this is easy to interpret. In our implementation called InterVENE, we use UMAP as dimension-reduction technique, because of its good run time performance and data structure preservation (cf. Section 2). An example of the visualizations is shown in Figure 4.

The user can select one of the visualizations (i.e. one of the layers at a certain training epoch), to inspect this projection in more detail. The selected projection x' is then used in an **interactive visualization** to show the patterns the machine learning model implicitly learned to best solve the prediction task. We visualize the following information about the machine learning model: i) an approximation of the learned feature space by the model x' (using 2d-position as visual channel), ii) the prediction of the model $\hat{y}$ (using color hue as visual channel), iii) whether the decision is a true positive, a false positive, a true negative or a false negative (using shape as visual channel). As shown in Figure 2, the visualization allows for interactive selection of subgroups of interest by either lasso selection with a mouse and/or choosing pre-selected categories, such as true and false positives or positives. The advantage of manual selection compared to automated clustering is that users can generate explanations about subgroups of data points where they have specific interest in. Data instances in these subgroups implicitly get assigned labels z indicating to which selection they belong. The *difference between the selected subgroups* is then explained by training an interpretable machine learning model to approximate the subgroup labels z based on the input features x. InterVENE uses a decision tree that classifies the selected data points to one of the selected subgroups. An example of this visualization and the explanation is shown in Figure 3.

4 Case Study and Data Set

InterVENE can be applied to any machine learning model that generates embeddings trained on any labeled dataset, since UMAP has no computational restrictions on embedding dimension [16] and the underlying techniques of InterVENE are generally applicable. To show the importance and benefits of InterVENE, we perform an exploratory study on a medical case where interpretability is critical. We use InterVENE to better understand a neural network predicting the neurological outcome of comatose patients after cardiac arrest. These predictions may prevent futile care in patients with a poor prognosis, as discussed in Section 2. Our structured dataset contains features from continuous EEG recordings of 518 prospectively collected adult patients who were comatose after cardiac arrest (Glasgow Coma Scale score $<$8) and admitted to the ICU of the Medisch Spectrum Twente (June 2010-May 2017) or Rijnstate hospital (June 2012-April 2017). Details have been described previously by Hofmeijer et al. [9].

The primary outcome measure was the Glasgow-Pittsburgh Cerebral Performance Category (CPC) score at 6 months, dichotomized as good (CPC 1 or 2, no or moderate neurological deficits with independence in activities of daily living) and poor (CPC score $\geq$ 3, major disability, coma, or death). The CPC scores were obtained prospectively by telephone follow-up with the patient or patient's legal representative. Part of the data is used in work of [28] and [19].

EEG features We extracted 42 qEEG features from each 10 second EEG segment at 24 hours after arrest to quantify 5 minute EEG epochs, broadly grouped into three domains: (i) time domain features capturing time varying amplitude information of the EEG signal; (ii) frequency domain features that capture key EEG patterns in different sub-bands in the spectrogram; and (iii) entropy domain features providing measures of complexity and randomness of the EEG signal. The median values of features across all channels were averaged for each 5 minute EEG epoch resulting in 42 features per patient. We used the same features as described by [19], except for 2 entropy features due to long calculation times. The data is scaled to have zero mean and unit variance, such that it can be used as input for a neural network. The dataset was complemented by a smaller set of features as described in [28] which are easier to interpret by the neurologist. These features are only used in an extra explanation, as discussed in Section 6.1.

5 Experimental Setup

Neural Network Architecture Since InterVENE can be used on any neural network architecture, the aim of this paper is not to train the best neural network for postanoxic coma classification, but to find a reasonable well-performing model which we want to explain. To find such a model, we optimized hyperparameters using grid search and stratified 5-fold cross validation on a training set (80% of the dataset) for a simple feedforward ANN. The following sets of hyperparameters were investigated by grid-search: #hidden layers $\{1, 2, 3\}$, #nodes in a hidden layer[4]: $\{5, 10, 20\}$, dropout: $\{$yes with $p = 0.5$, no$\}$, learning rate: $\{0.1, 0.01\}$, weight initialization: $\{\mathcal{N}(0, 1), \mathcal{N}(0, 0.1)\}$. To limit the number of networks to train, we fixed the activation function to ReLu (and Sigmoid in the output layer) and used the Adam optimizer. The most accurate network was a network with 2 hidden layers, with 20 hidden nodes in each hidden layer, dropout with $p = 0.5$, learning rate of 0.01 and weight initialization of $\mathcal{N}(0, 0.1)$. Using a hold out set, we decided to stop training after 100 epochs. Our network had an accuracy of 0.79 when using a threshold of 0.5 and AUC = 0.88. Because of a very high cost of error, prediction performance for poor outcome is in medical literature usually measured in recall (sensitivity) at 100% specificity. Our network has a recall of 0.34 for poor outcome at 100% specificity, outperforming the 0.32 of [28] and lower than state-of-the-art 0.44 [19], showing that our network is reasonably good.

Hyperparameters InterVENE For UMAP, we mainly use the default settings of the UMAP python package (v0.3). However, we set the number of neighbours n to 10 (default is 15) to more accurately catch the detailed manifold structure [16], and tuned the visualization by setting the distance metric to 'correlation'. For the decision tree, we use the default parameters of the scikit-learn decision tree package (v0.20.03). To improve interpretability and prevent overfitting, we set the maximum depth to 3 (but this parameter can be changed by the user), require a minimum number of 5 samples in each leaf and prune the tree such that a node is not split when all leaves have the same class label.

[4] with the restriction that the number of hidden nodes in a layer cannot exceed the number of hidden nodes in the previous layer

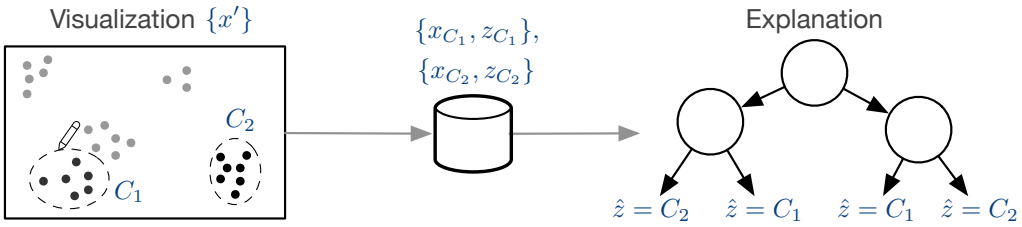


Figure 2. Overview of the interactive explanation. A user selects two groups of interest in visualization x', getting labels $z = C_1$ and $z = C_2$. Based on the input features x_{C_1}, x_{C_2}, a decision tree is trained to predict the corresponding clusters $\hat{z}$. The learnt decision tree thus explains the difference between the two selected clusters.

6 Experiments

We performed two experiments: (i) we elicited which domain knowledge can be obtained using InterVENE (see Section 6.1) and (ii) investigated which understanding about the neural networks in terms of error type and training process can be gained (see Section 6.2). We used the *constructive interaction* evaluation protocol in which two participants naturally communicate and collaborate in trying to solve predetermined tasks [18]. One participant in our study is a machine learning expert knowledgable about the projection method and neural networks. The other participant is a neurologist from Medisch Spectrum Twente with a full understanding of the medical dataset. Furthermore, we compare whether the discovered knowledge is in correspondence with existing medical literature (extracted knowledge validation).

6.1 Experiment 1: Neural Visualization and Explanation for Knowledge Discovery

We let the neurologist compare its domain knowledge with the results from our visualization and explanations. For this, the neurologist selected the visualization of the embedding of the last hidden layer of the final ANN shown in Figure 3(a). Tasks for the constructive interaction study consist of two components: 1) Interpreting the visualization, based on shape and colors; 2) Interpreting the explanation: selecting clusters and comparing the learnt decision tree with domain knowledge.

When looking at the visualization in Figure 3(a), both participants clearly see that the ANN is able to identify between comatose patients with a predicted good neurological outcome (yellow-green, 'cluster 2') and a predicted poor outcome (purple, 'cluster 1'). To get more insight, the neurologist uses InterVENE to manually select these two clusters, after which a decision tree is trained to explain the difference between these clusters. The learnt decision tree (not shown here), with a depth of 3, classifies each instance in the dataset as one of the two clusters with an accuracy of 0.913. The top feature in the tree is 'Hilbert burst'. The neurologist finds this decision tree plausible, since burst suppression (an EEG pattern in which neural activity with high amplitudes alternates with quiescence) is characteristic for an inactivated brain [10]. To evaluate the results, the neurologist visually inspects a few EEG recordings from each cluster. The neurologist clearly sees differences between poor and good outcome and would have made the same prediction as our neural network did.

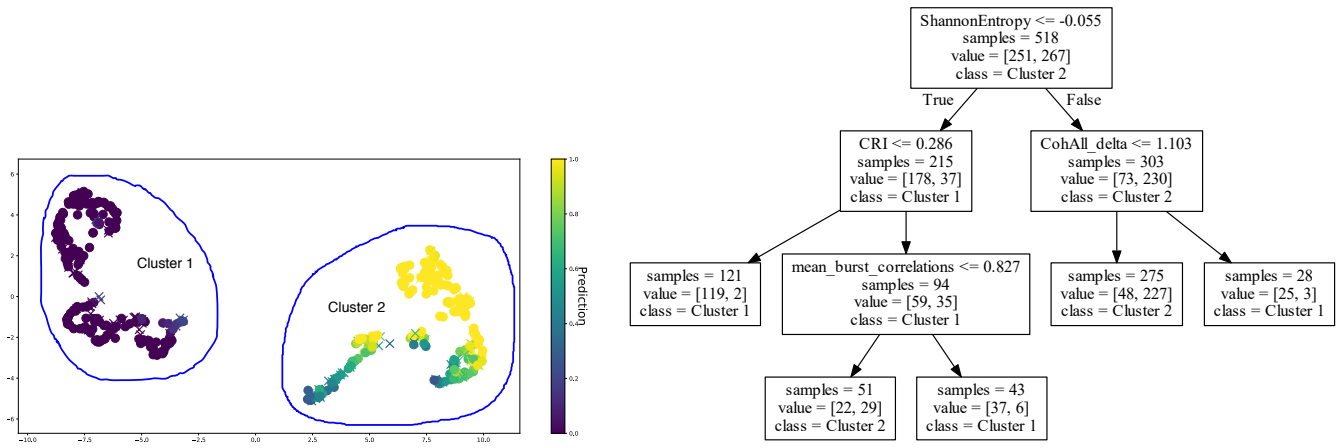
Based on the visualization in Figure 3(a), the neurologist is interested in studying the two sub-clusters within 'cluster 1', indicating that the ANN has learnt two different types of comatose patients that are expected to have a poor outcome. To explain this difference, the participants use the lasso selector of InterVENE to manually select these clusters. The resulting decision tree (not shown here) has a depth of 2 and an accuracy of 0.912 to classify an instance to one of the two purple clusters, with 'skewness' as top node. For evaluation of the result, a few EEGs from each subcluster are selected and analysed to evaluate whether the neurologist could see this same difference. Differentiation within 'cluster 1' was beyond visual assessment since skewness is a feature that cannot directly be read from an EEG; it can only be calculated.

Since manual classification is done by visual inspection of the EEG, features that cannot be directly read from an EEG are not used by neurologists in practice (although experiments like these might change this). This makes interpreting the features used in the decision tree more difficult. To solve this issue, we trained a second decision tree that only uses features that are easily interpretable by domain experts. Thus, while the visualizations are still based on the embeddings of an ANN trained on the 42 original features, the decision tree is only learnt from a dataset with 11 easy-to-interpret features (from the same patients). The decision tree shown in Figure 3 shows that Shannon entropy is the most important feature to distinguish between poor outcome ('cluster 1') and good outcome ('cluster 2'). This corresponds with existing literature which showed that Shannon entropy has the highest individual feature contribution for comatose patients 24 hours after cardiac arrest [28].

6.2 Experiment 2: Neural Visualization for Network Interpretation

In this experiment, we evaluate to what extent our approach aids neural network interpretation. We consider the visualizations of both hidden layers, and compare the visualizations of embeddings trained over time (after 10, 50 and 100 epochs). An overview with all the visualizations is shown in Figure 4. We defined the following tasks:
(i) See how the the neural network trains over time by comparing the visualizations of various epochs.
(ii) Interpret the relevance of hidden layers by comparing the visualization of deeper layers with visualizations of earlier layers.
(iii) Understand where the neural networks makes mistakes by analysing and comparing True Positives/Negatives with False Positives/Negatives.

Visualizations over time InterVENE shows how the artificial neural network (ANN) learns over time to give users a better understanding of the training process. From the top images in Figure 4, the two clusters indicate that the ANN is already able to distinguish between two groups of patients after training for 10 epochs. However, the ANN still makes many classification mistakes when trained for only 10 epochs. The lack of yellow and the presence of purple seem to indicate that the network quickly learns to recognize poor outcome, but is not confident yet about good outcomes. This is improved after 50 epochs, when the visualization clearly shows a yellow cluster



(a) Interactive Visualization (b) Explanation

Figure 3. UMAP visualization of the embedding of the 2nd hidden layer of the ANN, trained for 100 epochs. Each data point is a comatose patient. Color indicates the prediction. A circle indicates a correct classification; a cross indicates an incorrect classification (with prediction threshold 0.5). Users can interactively select 2 clusters of interest in the visualization using a lasso selector (blue lines). The difference between the selected clusters is explained by a decision tree, using only interpretable features (accuracy of 0.844). Each splitting node shows the binary expression, total number of samples, division of samples and class label.

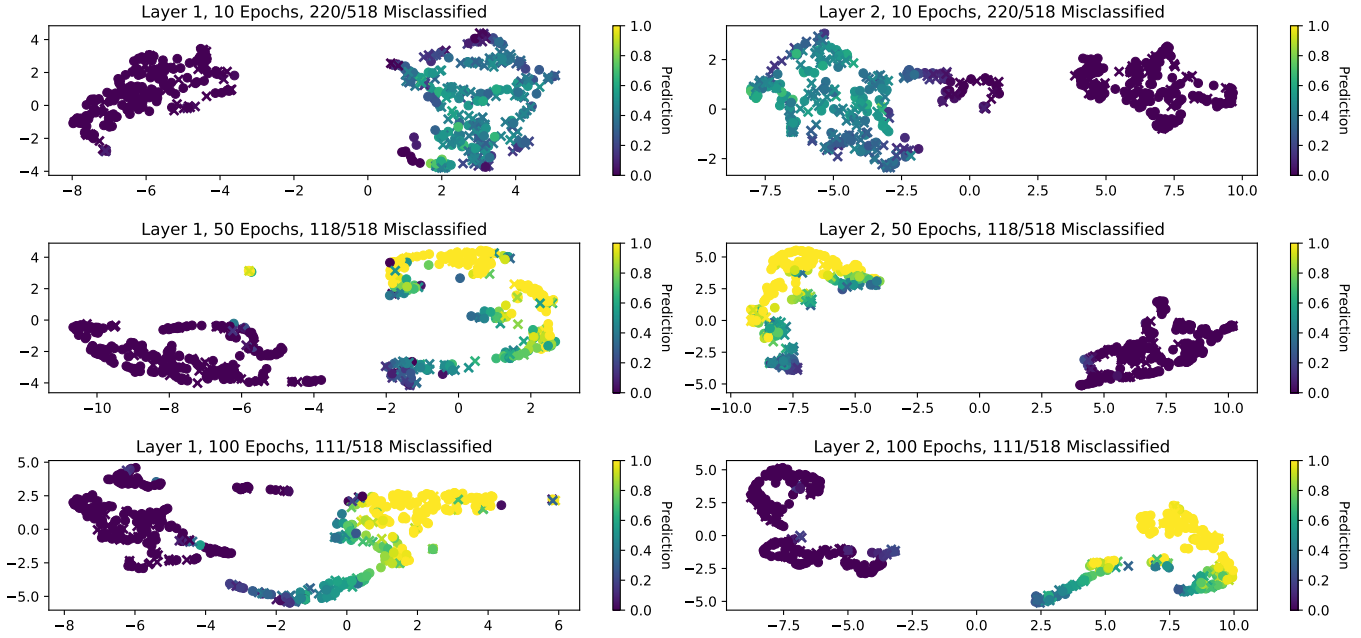


Figure 4. Visualized embeddings of two hidden layers over time. The left column shows the embedding of the first hidden layer, after the ANN is trained for 10, 50 resp. 100 epochs. The right column shows the embedding of the second hidden layer after 10, 50 resp. 100 epochs. The number of misclassifications indicates the number of incorrect classifications when a threshold of 0.5 is used to label the output of the ANN.

and a purple cluster. After 100 epochs, the second layer can clearly distinguish between good and poor outcome. Furthermore, it can better distinguish between yellow (confident about good outcome) and green (not confident about good outcome). Both the machine learning expert and domain expert found InterVENE useful to better understand the learning process of the ANN.

Relevance of hidden layers The participants noticed during the constructive interaction study that the visualizations of the first and second hidden layer do not differ substantially. Having comparable visualizations from different layers could indicate redundancy. Our grid-search found that a 2-layered model performed best but didn't tell how worse a 1-layer model would have been. To test whether the second layer is (almost) redundant, we trained an ANN with only one hidden layer, and compared its accuracy with the original 2-layered ANN. Whereas our ANN with 2 hidden layers misclassified 111 patients, the ANN with 1 hidden layer misclassified 120 patients. Since the increase is rather small, it confirms the hypothesis that the added value of the second layer is positive, but limited. This shows that InterVENE could be used for neural network pruning.

Analysing problematic instances InterVENE can also act as a starting point to further explore the training data and more quickly identify problematic training instances. As shown in Figure 5, users

can select a data subset such as true positives or false negatives using buttons. This gives more insights in the mistakes the ANN makes. When looking at the false negatives (Fig. 5), it is surprising to see that the ANN is very confident that some patients will have a poor outcome (purple) although in reality they had a good outcome. The neurologist is interested in this incorrect purple cluster, since a neural network incorrectly predicting a poor outcome (e.g. meaning that treatment can stop) can have severe consequences. InterVENE can explain the difference between e.g. true negatives and false negatives by learning a decision tree to distinguish between these two clusters. However, in our experiments the decision tree algorithm did not produce a decision tree that could distinguish between false negatives and true negatives. This means that patients with similar feature values in the dataset have different outcomes, which would explain why the ANN had difficulty to predict the correct outcome. This relevant information shows that some patients are not distinguishable, indicating that we might need to group them in a new class 'no safe prediction can be made'. We leave this 3-class prediction problem for future work.

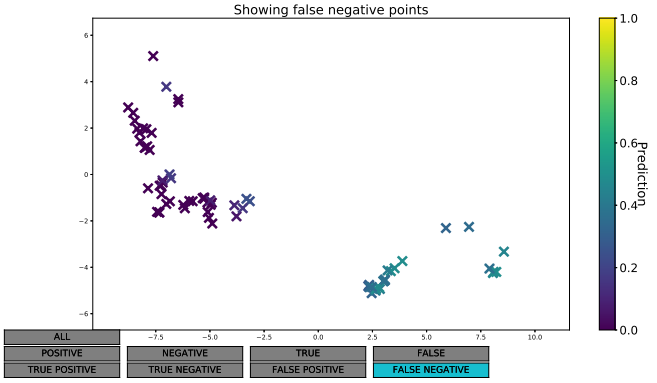


Figure 5. Screenshot of InterVENE. Button is used to only visualize false negatives, after which subgroup(s) can be selected by the user.

7 Discussion

InterVENE allows its users to interactively select groups of data points. However, literature raises concerns for clustering with t-SNE, which are salient for clustering the results of UMAP. In both methods, nearest neighbours are mostly preserved but distances between clusters and densities are not preserved well [24]. We therefore do not train an explainable method to predict the projected coordinates of an instance, but rather require it to only predict to which cluster a data instance belongs. However, users should still take these concerns into account when interpreting a UMAP visualization. Furthermore, although we applied InterVENE to only one dataset, we expect that InterVENE will perform well on other datasets. Since InterVENE let users select clusters instead of single datapoints, the quality of the visualization should not be impacted by the number of instances in the dataset. However, the number of features in a dataset might influence the quality of the explanation. Parameter tuning (either manual or automated) could ensure that an explanation is both accurate and interpretable (e.g. setting a max depth of the decision tree). Besides, InterVENE currently only allows 2-dimensional visualizations. This implementation can however easily be adapted, since UMAP supports higher dimensional visualizations.

8 Conclusion

Various projection techniques exist to visualize neural network embeddings. Since these visualizations are difficult to interpret, we presented an approach to *explain* these visualizations to aid knowledge discovery and network interpretation. We showed with a case study that users can get more insight in such a visualization by interactively selecting two subsets and comparing them with an interpretable, predictive model. Our implementation, called InterVENE, was applied to a structured dataset containing features about EEGs of comatose patients and is evaluated by a machine learning expert and domain expert (neurologist). An artificial neural network was trained to predict whether a patient would have a good outcome (e.g. wake up) or a poor outcome (e.g. further treatment is futile). After visualizing the neural embeddings, user can interact with InterVENE to generate a personalized decision tree which explains the difference between two user-selected subsets from the visualization. Our case study on comatose patients confirmed that InterVENE can successfully extract knowledge from a neural network. This knowledge was valuable for the neurologist and the explanations seemed plausible when compared with existing neurological domain knowledge. InterVENE also visualizes neural embeddings while the network is trained over time. Our experiments showed that this gives both domain experts and machine learning experts an idea of how the neural network is learning. Moreover, we showed that InterVENE can be used to judge the relevance of a hidden layer, by comparing the visualized embeddings of different hidden layers. For future work, we would like to apply InterVENE to other case studies, and train a network on raw EEG data and extract the learnt patterns from the network by learning a decision tree with hand-made features.

ACKNOWLEDGEMENTS

The authors would like to thank Duc Lê Trân Anh Dúc for improving the implementation of InterVENE.

REFERENCES

[1] Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, et al., 'Tensorflow: Large-scale machine learning on heterogeneous distributed systems', *arXiv preprint arXiv:1603.04467*, (2016).

[2] Etienne Becht, Leland McInnes, John Healy, Charles-Antoine Dutertre, Immanuel WH Kwok, Lai Guan Ng, Florent Ginhoux, and Evan W Newell, 'Dimensionality reduction for visualizing single-cell data using UMAP', *Nature Biotechnology*, **37**(1), 38, (2019).

[3] Matthew Brehmer, Michael Sedlmair, Stephen Ingram, and Tamara Munzner, 'Visualizing dimensionally-reduced data: Interviews with analysts and a characterization of task sequences', in *Proc. Workshop on Beyond Time and Errors: Novel Evaluation Methods for Visualization*, pp. 1–8. ACM, (2014).

[4] Wei Dong, Charikar Moses, and Kai Li, 'Efficient k-nearest neighbor graph construction for generic similarity measures', in *Proceedings of the 20th international conference on World wide web*, pp. 577–586. ACM, (2011).

[5] Finale Doshi-Velez and Been Kim, 'Towards a rigorous science of interpretable machine learning', *arXiv preprint arXiv:1702.08608*, (2017).

[6] L. H. Gilpin, D. Bau, B. Z. Yuan, A. Bajwa, M. Specter, and L. Kagal, 'Explaining Explanations: An Approach to Evaluating Interpretability of Machine Learning', *ArXiv e-prints*, (May 2018).

[7] Riccardo Guidotti, Anna Monreale, Salvatore Ruggieri, Franco Turini, Fosca Giannotti, and Dino Pedreschi, 'A survey of methods for explaining black box models', *ACM Comput. Surv.*, **51**(5), 93:1–93:42, (2018).

[8] G. Hinton, O. Vinyals, and J. Dean, 'Distilling the Knowledge in a Neural Network', *ArXiv e-prints*, (March 2015).

[9] J Hofmeijer, T.M.J. Beernink, F.H. Bosch, A Beishuizen, M.C. Tjepkema-Cloostermans, and M.J.A.M. van Putten, 'Early EEG contributes to multimodal outcome prediction of postanoxic coma', *Neurology*, **85**, 1–7, (2015).

[10] Jeannette Hofmeijer, Marleen C Tjepkema-Cloostermans, and Michel JAM van Putten, 'Burst-suppression with identical bursts: a distinct eeg pattern with poor outcome in postanoxic coma', *Clinical Neurophysiology*, **125**(5), 947–954, (2014).

[11] Eser Kandogan, 'Just-in-time annotation of clusters, outliers, and trends in point-based data visualizations', in *Proc. IEEE VAST*, pp. 73–82. IEEE, (2012).

[12] Klas Leino, Linyi Li, Shayak Sen, Anupam Datta, and Matt Fredrikson, 'Influence-directed explanations for deep convolutional networks', *CoRR*, **abs/1802.03788**, (2018).

[13] Xin Li, Ondrej E Dyck, Mark P Oxley, Andrew R Lupini, Leland McInnes, John Healy, Stephen Jesse, and Sergei V Kalinin, 'Manifold learning of four-dimensional scanning transmission electron microscopy', *npj Computational Materials*, **5**(1), 5, (2019).

[14] S. Liu, D. Maljovec, B. Wang, P. Bremer, and V. Pascucci, 'Visualizing high-dimensional data: Advances in the past decade', *IEEE Trans Vis Comput Graph*, **23**(3), 1249–1268, (March 2017).

[15] Laurens van der Maaten and Geoffrey Hinton, 'Visualizing data using t-SNE', *J. Mach. Learn. Res*, **9**(Nov), 2579–2605, (2008).

[16] Leland McInnes, John Healy, and James Melville, 'Umap: Uniform manifold approximation and projection for dimension reduction', *arXiv:1802.03426*, (2018).

[17] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean, 'Distributed representations of words and phrases and their compositionality', in *Advances in neural information processing systems*, pp. 3111–3119, (2013).

[18] Naomi Miyake, 'Constructive interaction and the iterative process of understanding', *Cognitive Science*, **10**(2), 151–177, (1986).

[19] Sunil B. Nagaraj, Marleen C. Tjepkema-Cloostermans, Barry J. Ruijter, Jeannette Hofmeijer, and Michel J.A.M. van Putten, 'The revised Cerebral Recovery Index improves predictions of neurological outcome after cardiac arrest', *Clinical Neurophysiology*, **129**(12), 2557–2566, (2018).

[20] Nicolas Papernot and Patrick D. McDaniel, 'Deep k-nearest neighbors: Towards confident, interpretable and robust deep learning', *CoRR*, **abs/1803.04765**, (2018).

[21] Paulo E Rauber, Samuel G Fadel, Alexandre X Falcao, and Alexandru C Telea, 'Visualizing the hidden activity of artificial neural networks', *IEEE Trans Vis Comput Graph*, **23**(1), 101–110, (2017).

[22] Ariella Richardson and Avi Rosenfeld, 'A survey of interpretability and explainability in human-agent systems', *Proc. Workshop on Explainable Artificial Intelligence, IJCAI*, 137–143, (2018).

[23] Andrew Slavin Ross, Michael C. Hughes, and Finale Doshi-Velez, 'Right for the right reasons: Training differentiable models by constraining their explanations', in *Proc. of the 26th Int. Joint Conf. on Artificial Intelligence, IJCAI-17*, pp. 2662–2670, (2017).

[24] Erich Schubert and Michael Gertz, 'Intrinsic t-stochastic neighbor embedding for visualization and outlier detection', in *Proc. Conf. Similarity Search and Applications*, pp. 188–203. Springer, (2017).

[25] Christin Seifert, Aisha Aamir, Aparna Balagopalan, Dhruv Jain, Abhinav Sharma, Sebastian Grottel, and Stefan Gumhold, 'Visualizations of deep neural networks in computer vision: A survey', in *Transparent Data Mining for Big and Small Data*, 123–144, Springer, Cham, (2017).

[26] Daniel Smilkov, Nikhil Thorat, Charles Nicholson, Emily Reif, Fernanda B Viégas, and Martin Wattenberg, 'Embedding projector: Interactive visualization and interpretation of embeddings', *NIPS 2016 Workshop on Interpretable Machine Learning in Complex Systems*, (2016).

[27] Julian Stahnke, Marian Dörk, Boris Müller, and Andreas Thom, 'Probing projections: Interaction techniques for interpreting arrangements and errors of dimensionality reductions', *IEEE Trans Vis Comput Graph*, **22**(1), 629–638, (2016).

[28] Marleen Tjepkema-Cloostermans, Jeannette Hofmeijer, Albertus Beishuizen, Harold W Hom, Michiel J Blans, Frank H Bosch, and Michel JAM Van Putten, 'Cerebral recovery index: reliable help for prediction of neurologic outcome after cardiac arrest', *Critical care medicine*, **45**(8), e789–e797, (2017).

[29] Marleen Tjepkema-Cloostermans, Catarina Lourence, B.J. Ruijter, A. Beishuizen, Gea Drost, Frank H. Bosch, Francois H Kornips, J. Hofmeijer, and M.J.A.M. van Putten, 'Outcome prediction in postanoxic coma with deep learning', *Critical Care Medicine*, in press, (2019).

[30] Kai Xu, Dae Hoon Park, Chang Yi, and Charles Sutton, 'Interpreting deep classifier by visual distillation of dark knowledge', *arXiv preprint arXiv:1803.04042*, (2018).

[31] Yongxin Yang, Irene Garcia Morillo, and Timothy M. Hospedales, 'Deep neural decision trees', *CoRR*, **abs/1806.06988**, (2018).