Semester Thesis: Measuring leakage in Concept-based methods

Mikael Makonnen, Moritz Vandenhirtz

November 19, 2024

1 Background & Motivation

Concept Bottleneck Models Concept bottleneck models (CBM) Koh et al. (2020); Lampert et al. (2009); Kumar et al. (2009) are a simple class of interpretable neural networks typically trained on data points $(\boldsymbol{x}, \boldsymbol{c}, \boldsymbol{y})$, comprising the covariates $\boldsymbol{x} \in \mathcal{X}$ and targets $\boldsymbol{y} \in \mathcal{Y}$ additionally annotated by the concepts $\boldsymbol{c} \in \mathcal{C}$. Consider a neural network $f_{\boldsymbol{\theta}}$ parameterised by $\boldsymbol{\theta}$ and a slice $\langle g_{\boldsymbol{\psi}}, h_{\boldsymbol{\phi}} \rangle$ Leino et al. (2018) s.t.

$$f_{\boldsymbol{\theta}}(\boldsymbol{x}) = g_{\boldsymbol{\psi}}(h_{\boldsymbol{\phi}}(\boldsymbol{x})) \tag{1}$$

for all $\boldsymbol{x} \in \mathcal{X}$, where $\hat{\boldsymbol{y}} := f_{\boldsymbol{\theta}}(\boldsymbol{x}) = g_{\boldsymbol{\psi}}(h_{\boldsymbol{\phi}}(\boldsymbol{x}))$ denote the output of the network, *i.e.* the predicted targets. CBMs enforce a concept bottleneck $\hat{\boldsymbol{c}} := h_{\boldsymbol{\phi}}(\boldsymbol{x})$: the model's final output depends on the covariates $\boldsymbol{x}$ solely through the predicted concepts $\hat{\boldsymbol{c}}$. Thus, in addition to the target prediction loss applied to the final output, $h_{\boldsymbol{\phi}}(\cdot)$ is trained to predict the ground-truth concept values.

Interpretability The interpretability of CBMs is achieved by the set of high-level, human-understandable concepts. Often, these are C binary-valued attributes, i.e. $\mathcal{C} = \{0, 1\}^C$ that can be easily detected from the covariates $\boldsymbol{x}$ and are predictive of the targets $\boldsymbol{y}$. Although CBMs make no assumptions on (anti)causal relationships among $\boldsymbol{x}$, $\boldsymbol{c}$, and $\boldsymbol{y}$, they implicitly assume that concepts $\boldsymbol{c}$ are a sufficient statistic Yeh et al. (2020) for predicting $\boldsymbol{y}$ based on $\boldsymbol{x}$ Havasi et al. (2022); Marcinkevičs et al. (2024), i.e. $\boldsymbol{y} \perp\!\!\!\perp \boldsymbol{x} \mid \boldsymbol{c}$.

Leakage Leakage is an instance of shortcut learning (Geirhos et al., 2020). Margeloiu et al. (2021); Mahinpei et al. (2021); Havasi et al. (2022) show that leakage occurs in cases where the conditional independence assumption does not hold. The distribution of the predicted concept values encodes more information than solely the probability of concept presence. This additional information can then be exploited by the classifier $g_{\boldsymbol{\psi}}(\cdot)$. This is an issue since the predicted concept values encode information different from the human-understandable concepts, thus, prohibiting the interpretation of the predicted probability as probability of concept presence. Mahinpei et al. (2021) show that even if the predicted concepts are not soft (*i.e.* $\boldsymbol{c} \in [0, 1]$) but hard (*i.e.* $\boldsymbol{c} \in \{0, 1\}$), leakage happens, albeit weaker. Therefore, any perception of interpretability for standard CBMs is void if $\boldsymbol{y} \perp\!\!\!\perp \boldsymbol{x} \mid \boldsymbol{c}$ is not fulfilled, which is often the case in real-world problems. Examples of works (unintentionally) committing this fallacy are Espinosa Zarlenga et al. (2022); Marconato et al. (2022); Ismail et al. (2023). To understand how strongly the interpretability of concept probabilities is restricted, we need a metric that is able to measure the leakage within these concept embeddings.

2 Related Work

To measure leakage, Zarlenga et al. (2023) propose metrics that estimate the degree of excessive information with respect to other concepts, which they call impurity. To resolve leakage, Margeloiu et al. (2021) recommend using the *independent* training procedure with hard concepts. However, this comes at the cost of decreasing performance since the encoder and predictor head can not communicate anymore. Thus, Havasi et al. (2022) propose to include a hard side-channel, in which the additional information can be learned explicitly, as well as an autoregressive structure over the hard concept predictions, such that their correlations can be captured. At intervention time, they use importance-weighted MCMC sampling to implicitly learn the effect of a concept intervention on the other concepts. It will be interesting to see whether their approach fully eradicates leakage.

3 Methods

Consider a neural network $NN_{\boldsymbol{\theta}}$ parameterised by $\boldsymbol{\theta}$ and a slice $\langle g_{\boldsymbol{\psi}}, h_{\boldsymbol{\phi}} \rangle$ Leino et al. (2018) s.t.

$$NN_{\boldsymbol{\theta}}\left(\boldsymbol{x}\right) = g_{\boldsymbol{\psi}}\left(h_{\boldsymbol{\phi}}\left(\boldsymbol{x}\right)\right) \tag{2}$$

For sake of intuition, think of it as a CBM, where $\boldsymbol{z} = h_{\boldsymbol{\phi}}(\boldsymbol{x}) = \hat{\boldsymbol{c}}$ is trained via the prediction of concepts, but this formulation allows for a more general interpretation.

What we are interested in for leakage, is the information contained within $\boldsymbol{z}$, which is informative for the label $\boldsymbol{y}$ but independent/non-informative of concepts $\boldsymbol{c}$:

$$I(\boldsymbol{z}; \boldsymbol{y} \mid \boldsymbol{c}) = H(\boldsymbol{y} \mid \boldsymbol{c}) - H(\boldsymbol{y} \mid \boldsymbol{z}, \boldsymbol{c})$$

Estimating $H(\boldsymbol{y} \mid \boldsymbol{c})$ and $H(\boldsymbol{y} \mid \boldsymbol{z}, \boldsymbol{c})$ is the goal of this thesis. A straightforward approximation is

$$H(\boldsymbol{y} \mid \boldsymbol{z}, \boldsymbol{c}) = \mathbb{E}[-\log p(\boldsymbol{y} \mid \boldsymbol{z}, \boldsymbol{c})] \approx -\frac{1}{N}\sum_{i=1}^{N} \log g_{a,\boldsymbol{\psi}}\left(h_{\boldsymbol{\phi}}\left(\boldsymbol{x}_i\right), \boldsymbol{c}_i\right)_{y_i}, \tag{3}$$

$$H(\boldsymbol{y} \mid \boldsymbol{c}) = \mathbb{E}[-\log p(\boldsymbol{y} \mid \boldsymbol{c})] \approx -\frac{1}{N}\sum_{i=1}^{N} \log g_{b,\boldsymbol{\psi}}\left(\boldsymbol{c}_i\right)_{y_i}, \tag{4}$$

where $g_{a,\boldsymbol{\psi}}$ and $g_{b,\boldsymbol{\psi}}$ are two classifiers trained to predict $\boldsymbol{y}$ from $\boldsymbol{z}$, $\boldsymbol{c}$ and from $\boldsymbol{c}$, respectively.

4 Deliverables

The following subsections are loosely listed in the order of execution, but especially for the later packages, the order might vary, or we might not do them at all.

4.1 Work package 1 (Literature Review)

Perform an extensive literature review on the topic. Some of the questions we are interested in are the following:

- Is our leakage definition sound? Are there established metrics in the shortcut learning literature that we might be able to use?
- Is the proposed way of measuring the mutual-information-based leakage as the difference of the two conditional entropies the best and only approach or are there alternatives?
- Which methods g lend themselves to estimating the entropy?
- Is it an issue that $\boldsymbol{z} = h_{\boldsymbol{\phi}}(\boldsymbol{x}_i)$ itself has been learned with the training data? That is, can we reuse the training data to train $g_{a,\boldsymbol{\psi}}(h_{\boldsymbol{\phi}}(\boldsymbol{x}_i), \boldsymbol{c}_i)$ or is $\boldsymbol{z}$ overfitted and not generalizable to validation/test? MV: Currently, I think should be okay, because the classifier head also takes them as input and still works?
- Should g_a and g_b be separate estimators, or is there a smart way to combine them in one estimator to ensure that the estimated $H(\boldsymbol{y} \mid \boldsymbol{c}) \geq H(\boldsymbol{y} \mid \boldsymbol{z}, \boldsymbol{c})$?

Note that the questions do not need to be answered, but rather multiple ideas can be gathered, which can then be tested against each other in the next step.

4.2 Work package 2 (Synthetic Experiment(s))

The goal of this package is to narrow down the space of potential approaches of measuring leakage, be it by varying the metric itself, or the approximators.

Design synthetic experiments where we have (near-)perfect supervision on the data-generating mechanism such that we know how much leakage would be expected. Then, we can evaluate the multiple ideas & approaches obtained in Work Package 1. For example, we can train a jointly trained CBM with different weights λ for concept encoder and predictor to vary the amount of leakage. Alternatively/Additionally, as a first step, we don't have to train a CBM at all and can instead construct $\boldsymbol{z}$ in such a way that we know exactly how much information with respect to $\boldsymbol{c}$ and $\boldsymbol{y}$ is contained. The synthetic experiments can start directly from $\boldsymbol{z}$ instead of $\boldsymbol{x}$, such that we can design different $\boldsymbol{z}$ whose leakage (or at least their relative order) is known.

4.3 Work package 3 (Evaluating Concept-based Methods on Synthetic)

Given a working synthetic setup, we can start comparing existing concept-based methods against each other. Some methods would be: Joint Soft CBM with varying λ (Koh et al., 2020), Sequential and Independent CBM (Koh et al., 2020), Autoregressive CBM (Havasi et al., 2022), Concept Embedding Model (Espinosa Zarlenga et al., 2022), Stochastic Bottleneck Models (our paper, currently in submission). The code for all methods is available in our private repository, and we have an intuition how the methods should perform.

The end of this package provides a natural stopping point in which we could conclude the first "official semester thesis".

4.4 Work package 4 (From Synthetic to Real-World)

Given a working setup in the synthetic dataset, the next goal is to extend it to real-world datasets. As such, the goal will be to apply the developed metric to real-world datasets such as CUB, CelebA, CIFAR-10. Likely, it will require some adaptions and tuning in the metric approximations and approximators.

This work package concludes with the evaluation of aforementioned concept-based methods on these datasets.

4.5 Work package 5 (Further Steps)

Depending on the outcome of previous sections, there are different interesting paths that can be followed:

- Can the leakage measure be used as regularizer during training to de-leak jointly trained CBMs?
- We might want to replace $\boldsymbol{z}$ by $g_{\psi}(\boldsymbol{z})$ to indicate that leakage can exist in the embedding $\boldsymbol{z}$ as long as the classifier g_{ψ} does not pick up on it. New estimation methods need to be developed that capture which information the classifier picks up on. Formally, we would estimate

$$I(\hat{\boldsymbol{y}}; \boldsymbol{y} \mid \boldsymbol{c}) = H(\boldsymbol{y} \mid \boldsymbol{c}) - H(\boldsymbol{y} \mid \hat{\boldsymbol{y}}, \boldsymbol{c})$$

- In order to bound the leakage metric, we could normalize the mutual information $I(\boldsymbol{z}; \boldsymbol{y} \mid \boldsymbol{c})$ by the maximum possible information contained in the embedding, which corresponds to encoding all of $\boldsymbol{x}$, i.e., $I(\boldsymbol{x}; \boldsymbol{y} \mid \boldsymbol{c})$. It can be calculated as $I(\boldsymbol{x}; \boldsymbol{y} \mid \boldsymbol{c}) = H(\boldsymbol{y} \mid \boldsymbol{c}) - H(\boldsymbol{y} \mid \boldsymbol{x}, \boldsymbol{c})$, where the entropy $H(\boldsymbol{y} \mid \boldsymbol{x}, \boldsymbol{c})$ can be estimated from a third approximator $g_{c,\psi}$, trained to predict $\boldsymbol{y}$ from $\boldsymbol{x}$ and $\boldsymbol{c}$. This would result in the final leakage metric being $\frac{I(\boldsymbol{z};\boldsymbol{y}|\boldsymbol{c})}{I(\boldsymbol{x};\boldsymbol{y}|\boldsymbol{c})}$.
- Should g_a and g_b be separate estimators, or is there a smart way to combine them in one estimator to ensure that the estimated $H(\boldsymbol{y} \mid \boldsymbol{c}) \geq H(\boldsymbol{y} \mid \boldsymbol{z}, \boldsymbol{c})$?
- Writing a workshop paper with current results.

References

Mateo Espinosa Zarlenga, Pietro Barbiero, Gabriele Ciravegna, Giuseppe Marra, Francesco Giannini, Michelangelo Diligenti, Zohreh Shams, Frederic Precioso, Stefano Melacci, Adrian Weller, et al. Concept embedding models: Beyond the accuracy-explainability trade-off. In *Advances in Neural Information Processing Systems*, volume 35, pages 21400–21413, 2022.

Robert Geirhos, Jörn-Henrik Jacobsen, Claudio Michaelis, Richard Zemel, Wieland Brendel, Matthias Bethge, and Felix A Wichmann. Shortcut learning in deep neural networks. *Nature Machine Intelligence*, 2(11):665–673, 2020.

Marton Havasi, Sonali Parbhoo, and Finale Doshi-Velez. Addressing leakage in concept bottleneck models. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho, editors, *Advances in Neural Information Processing Systems*, 2022. URL https://openreview.net/forum?id=tglniD_fn9.

Aya Abdelsalam Ismail, Julius Adebayo, Hector Corrada Bravo, Stephen Ra, and Kyunghyun Cho. Concept bottleneck generative models, 2023.

Pang Wei Koh, Thao Nguyen, Yew Siang Tang, Stephen Mussmann, Emma Pierson, Been Kim, and Percy Liang. Concept bottleneck models. In Hal Daumé III and Aarti Singh, editors, *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 5338–5348, Virtual, 2020. PMLR. URL https://proceedings.mlr.press/v119/koh20a.html.

Neeraj Kumar, Alexander C. Berg, Peter N. Belhumeur, and Shree K. Nayar. Attribute and simile classifiers for face verification. In *2009 IEEE 12th International Conference on Computer Vision*, pages 365–372, Kyoto, Japan, 2009. IEEE. URL https://doi.org/10.1109/ICCV.2009.5459250.

Christoph H. Lampert, Hannes Nickisch, and Stefan Harmeling. Learning to detect unseen object classes by between-class attribute transfer. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, Miami, FL, USA, 2009. IEEE. URL https://doi.org/10.1109/CVPR.2009.5206594.

Klas Leino, Shayak Sen, Anupam Datta, Matt Fredrikson, and Linyi Li. Influence-directed explanations for deep convolutional networks. In *2018 IEEE International Test Conference (ITC)*. IEEE, 2018. URL https://doi.org/10.1109/test.2018.8624792.

Anita Mahinpei, Justin Clark, Isaac Lage, Finale Doshi-Velez, and Weiwei Pan. Promises and pitfalls of black-box concept learning models, 2021. URL https://doi.org/10.48550/arXiv.2106.13314. *arXiv:2106.13314*.

Ričards Marcinkevičs, Patricia Reis Wolfertstetter, Ugne Klimiene, Kieran Chin-Cheong, Alyssia Paschke, Julia Zerres, Markus Denzinger, David Niederberger, Sven Wellmann, Ece Ozkan, Christian Knorr, and Julia E. Vogt. Interpretable and intervenable ultrasonography-based machine learning models for pediatric appendicitis. *Medical Image Analysis*, 91:103042, 2024. URL https://www.sciencedirect.com/science/article/pii/S136184152300302X.

Emanuele Marconato, Andrea Passerini, and Stefano Teso. GlanceNets: Interpretable, leak-proof concept-based models, 2022. *arXiv:2205.15612*.

Andrei Margeloiu, Matthew Ashman, Umang Bhatt, Yanzhi Chen, Mateja Jamnik, and Adrian Weller. Do concept bottleneck models learn as intended?, 2021. URL https://doi.org/10.48550/arXiv.2105.04289. *arXiv:2105.04289*.

Chih-Kuan Yeh, Been Kim, Sercan Arik, Chun-Liang Li, Tomas Pfister, and Pradeep Ravikumar. On completeness-aware concept-based explanations in deep neural networks. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 20554–20565. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/ecb287ff763c169694f682af52c1f309-Paper.pdf.

Mateo Espinosa Zarlenga, Pietro Barbiero, Zohreh Shams, Dmitry Kazhdan, Umang Bhatt, Adrian Weller, and Mateja Jamnik. Towards robust metrics for concept representation evaluation. *arXiv preprint arXiv:2301.10367*, 2023.