Set-Valued Prediction in Hierarchical Classification with Constrained Representation Complexity (Supplementary Material)

Thomas Mortier[1] **Eyke Hüllermeier**[2] **Krzysztof Dembczyński**[3,4] **Willem Waegeman**[1]

[1]Dept. of Data Analysis and Mathematical Modelling, Ghent University, Coupure links 653, Ghent, Belgium
[2]Institute of Informatics, LMU Munich, Akademiestr. 7, Munich, Germany
[3]Institute of Computing Science, Poznań University of Technology, Piotrowo 2, Poznań, Poland
[4]Yahoo! Research, 770 Broadway, New York, USA

A PROOF OF THEOREM 1

We first prove an intermediate result.

Proposition 1. *For any class space $\mathcal{Y}$ and valid hierarchy $\mathcal{T}$ we have that:*

$$\forall i,j \in [K-1] : \mathcal{R}_{\mathcal{T}}^{(i)} \neq \mathcal{R}_{\mathcal{T}}^{(j)} \implies \mathcal{R}_{\mathcal{T}}^{(i)} \cap \mathcal{R}_{\mathcal{T}}^{(j)} = \emptyset \quad (1)$$

Proof. Let i, j with $\mathcal{R}_{\mathcal{T}}^{(i)} \neq \mathcal{R}_{\mathcal{T}}^{(j)}$ and assume that $\mathcal{R}_{\mathcal{T}}^{(i)} \cap \mathcal{R}_{\mathcal{T}}^{(j)} \neq \emptyset$. For $\hat{Y} \in \mathcal{R}_{\mathcal{T}}^{(i)} \cap \mathcal{R}_{\mathcal{T}}^{(j)}$, we know that:

$$R_{\mathcal{T}}(\hat{Y}) = i \Leftrightarrow \min_{\hat{V} \in \mathcal{S}_{\mathcal{T}}(\hat{Y})} |\hat{V}| = i \,,$$
$$R_{\mathcal{T}}(\hat{Y}) = j \Leftrightarrow \min_{\hat{V} \in \mathcal{S}_{\mathcal{T}}(\hat{Y})} |\hat{V}| = j \,,$$

and, hence, is only possible when $i = j$, which contradicts with the beginning of this proof. □

In order to prove Theorem 1, we need to show that the following conditions are met:

1. $\forall i,j \in [K-1] : \mathcal{R}_{\mathcal{T}}^{(i)} \neq \mathcal{R}_{\mathcal{T}}^{(j)} \implies \mathcal{R}_{\mathcal{T}}^{(i)} \cap \mathcal{R}_{\mathcal{T}}^{(j)} = \emptyset$
2. $\bigcup_{i \in [K-1]} \mathcal{R}_{\mathcal{T}}^{(i)} = \mathcal{P}(\mathcal{Y})$

The first condition is met due to Proposition 1. To show that the second condition is met, we need to prove that $\hat{Y} \in \bigcup_{i \in [K-1]} \mathcal{R}_{\mathcal{T}}^{(i)} \implies \hat{Y} \in \mathcal{P}(\mathcal{Y}) \wedge \hat{Y} \in \mathcal{P}(\mathcal{Y}) \implies \hat{Y} \in \bigcup_{i \in [K-1]} \mathcal{R}_{\mathcal{T}}^{(i)}$. We start by proving the first part, which follows trivially from the definition of a representation complexity class, as each set that belongs to a given representation complexity class must be element of $\mathcal{P}(\mathcal{Y})$. To prove the second part, it suffices to show that $\forall \hat{Y} \in \mathcal{P}(\mathcal{Y}) : \mathcal{S}_{\mathcal{T}}(\hat{Y}) \neq \emptyset$, or in other words, for each element $\hat{Y}$ in $\mathcal{P}(\mathcal{Y})$ there exists at least one $\hat{V} \subset \mathcal{V}_{\mathcal{T}}$ such that:

$$\bigcup_{v_i \in \hat{V}} v_i = \hat{Y} \,, \quad \bigcap_{v_i \in \hat{V}} v_i = \emptyset \,.$$

Note that each element $\hat{Y} \in \mathcal{P}(\mathcal{Y})$ can be represented by either a node in the hierarchy, the union of sets of leaf nodes in the hierarchy $\mathcal{T}$:

$$\hat{Y} = \bigcup_{c_i \in \hat{Y}} \{c_i\} \,,$$

or by a union of internal and/or leaf nodes. From this, it follows that $\mathcal{S}_{\mathcal{T}}(\hat{Y}) \neq \emptyset$ and $R_{\mathcal{T}}(\hat{Y}) = \min_{\hat{V} \in \mathcal{S}_{\mathcal{T}}(\hat{Y})} |\hat{V}| = i$, where i is lower bounded by one and upper bounded by $|\hat{Y}|$. Therefore, given the above, it follows that $\forall \hat{Y} \in \mathcal{P}(\mathcal{Y}) \,, \exists\, i \in [K-1] \,: \hat{Y} \in \mathcal{R}_{\mathcal{T}}^{(i)}$ which proves the second and last part of this proof.

B EXPERIMENTAL SETUP

We use a MobileNetV2 convolutional neural network [Sandler et al., 2018], pretrained on ImageNet [Deng et al., 2009], to obtain hidden representations for all image datasets. For the bacteria dataset, tf-idf representations are obtained by means of extracting 3-, 4- and 5-grams from the 16S rRNA sequences that were provided in the dataset [Fiannaca et al., 2018]. For the proteins dataset, tf-idf representations are obtained by considering 3-grams only. Furthermore, to comply with literature, the tf-idf representations are concatenated with functional domain encodings, which contain distinct functional and evolutional information about the protein sequence [Li et al., 2018]. Next, the obtained feature representations for the biological datasets are then passed through a single-layer neural net with 1000 output neurons and a ReLU activation function. We use the categorical cross-entropy loss by means of stochastic gradient descent with momentum, where the learning rate and momentum are set to $1e-5$ and 0.99, respectively. For the models without hierarchical factorization, we set the number of epochs to 2 and 20, for the Caltech and other datasets, respectively. For the models with hierarchical factorization, we use 4 and 30, respectively. We train all models end-to-end on a GPU, by using the PyTorch library [Paszke et al., 2017] and infrastructure with the following specifications:

Accepted for the 38th Conference on Uncertainty in Artificial Intelligence (UAI 2022).

- **CPU:** i7-6800K 3.4 GHz (3.8 GHz Turbo Boost) – 6 cores / 12 threads,
- **GPU:** 2x Nvidia GTX 1080 Ti 11GB + 1x Nvidia Tesla K40c 11GB,
- **RAM:** 64GB DDR4-2666.

Finally, we implemented the RTS and TOP-k algorithms in C++ by using the PyTorch C++ API [Paszke et al., 2017].

References

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *IEEE CVPR*, pages 248–255, 2009.

Antonino Fiannaca, Laura La Paglia, Massimo La Rosa, Giosue Lo Bosco, Giovanni Renda, Riccardo Rizzo, Salvatore Gaglio, and Alfonso Urso. Deep learning models for bacteria taxonomic classification of metagenomic data. *BMC Bioinformatics*, 19-S(7):61–76, 2018.

Yu Li, Sheng Wang, Ramzan Umarov, and et al. Deepre: sequence-based enzyme EC number prediction by deep learning. *BMC Bioinformatics*, 34(5):760–769, 2018.

Adam Paszke, Sam Gross, Soumith Chintala, and et al. Automatic differentiation in pytorch. In *NIPS-W*, 2017.

Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. Mobilenetv2: Inverted residuals and linear bottlenecks, 2018.