Large-scale zero-shot learning in the wild: classifying zoological illustrations

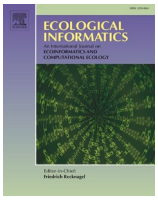
Stork, L.; Weber, A.; Herik, H.J. van den; Plaat, A.; Verbeek, F.J.; Wolstencroft, K.J.

Ecological Informatics 62 (2021) 101222

Contents lists available at ScienceDirect

Ecological Informatics

journal homepage: www.elsevier.com/locate/ecolinf



Large-scale zero-shot learning in the wild: Classifying zoological illustrations

Lise Stork [a,*], Andreas Weber [b], Jaap van den Herik [a,c], Aske Plaat [a], Fons Verbeek [a], Katherine Wolstencroft [a]

[a] Leiden Institute of Advanced Computer Science, Niels Bohrweg 1, 2333 CA Leiden, the Netherlands
[b] University of Twente, Drienerlolaan 5, 7522 NB Enschede, the Netherlands
[c] The Leiden Centre of Data Science, Leiden, the Netherlands

ARTICLE INFO

Keywords:
Zero-shot learning
Biodiversity
Natural history
Hierarchical learning
Fine-grained object recognition
Small samples

ABSTRACT

In this paper we analyse the classification of zoological illustrations. Historically, zoological illustrations were the modus operandi for the documentation of new species, an Currently, they serve as crucial sources for long-term ecological and biodiversity research. By employing computational methods for classification, the illustrations can be made amenable to research. Automated species identification is challenging due to the long-tailed nature of the data, and the millions of possible classes in the species taxonomy. Success commonly depends on large training sets with many examples per class, but images from only a subset of classes are digitally available, and many images are unlabelled, since labelling requires domain expertise. We explore zero-shot learning to address the problem, where features are learned from classes with medium to large samples, which are then transferred to recognise classes with few or no training samples. We specifically explore how distributed, multimodal background knowledge from data providers, such as the Global Biodiversity Information Facility (GBIF), iNaturalist, and the Biodiversity Heritage Library (BHL), can be used to share knowledge between classes for zero-shot learning. We train a prototypical network for zero-shot classification, and introduce fused prototypes (FP) and hierarchical prototype loss (HPL) to optimise the model. Finally, we analyse the performance of the model for use in real-world applications. The experimental results are encouraging, indicating potential for use of such models in an expert support system, but also express the difficulty of our task, showing a necessity for research into computer vision methods that are able to learn from small samples.

1. Introduction

Zero-shot learning (ZSL) aims to recognise objects whose instances have not yet been seen during training, based on semantic knowledge, e.g., attributes (Ferrari and Zisserman, 2007; Lampert et al., 2014), shared among seen and unseen classes. Datasets have been set up to facilitate progress in the field and demonstrate the possibilities and advantages of zero-shot learning (Lampert et al., 2014; Patterson and Hays, 2012; Wah et al., 2011).

We argue there is a need for research that analyses the performance of zero-shot learning models on complex real-world data, collected to fulfill a need within a certain domain, e.g., (Sumbul et al., 2018; Van Horn et al., 2018). Specifically data from domains where the solution space is large and complex, and obtaining labels for training is costly or simply not feasible. When algorithms are evaluated on highly imbalanced large-scale datasets, results are often poor. Xian et al. show that experiments of state-of-the-art zero-shot learning algorithms achieve only ~1.3% top-1 per-class accuracy on the 5000 least populated classes in ImageNet, and only ~0.4% top-1 accuracy for generalised zero-shot learning (GZSL) (Xian et al., 2019), where the classifier must choose the correct class from both seen *and* unseen classes.

In this paper we introduce and analyse an imbalanced, sparsely populated and hierarchical large-scale dataset for zero-shot learning. The dataset comes from the natural history domain, consists of 14,502 zoological illustrations of 7973 species from the animal kingdom, and is formed by consolidating data used and managed by the biodiversity research community. Automated species identification is a much researched problem within the computer vision and pattern recognition

* Corresponding author.
E-mail addresses: l.stork@vu.nl (L. Stork), a.weber@utwente.nl (A. Weber), h.j.vandenherik@law.leidenuniv.nl (J. van den Herik), a.plaat@liacs.leidenuniv.nl (A. Plaat), f.j.verbeek@liacs.leidenuniv.nl (F. Verbeek), k.j.wolstencroft@liacs.leidenuniv.nl (K. Wolstencroft).

https://doi.org/10.1016/j.ecoinf.2021.101222
Received 17 August 2020; Received in revised form 17 November 2020; Accepted 17 November 2020
Available online 30 January 2021

domain, but, to the best of our knowledge, no approaches have been described to deal with the wealth of detailed zoological illustrations (example shown in Fig. 1). Reasons could be that samples are small due to the nature of the data - many rare species have been depicted in small quantities - and because numerous institutions have yet to start with the digitisation of their collections (Drew et al., 2017). Ultimately, automated methods can assist biodiversity experts in the formation of a global picture of historical and current biodiversity, something that is crucial given the current biodiversity crisis (Hedrick et al., 2020).

1.1. Zoological illustrations

Historically, the habitus illustration - a scientific illustration of a species' physical appearance - was the most important medium to convey a species' characterising traits to other scientists. In illustrations, scientists are capable of delineating and highlighting minuscule details, often more so than photographs. Habitus illustrations were routinely and abundantly created and commonly served as examples for the description of newly discovered species, so-called holotypes. Additionally, they sometimes recorded the habitat or behaviour of an organism. Over the last 250 years, a large number of zoological species have been observed and documented this way, by means of expeditions to biodiverse areas worldwide.

Research into these scientific illustrations is complicated by several challenges. First, most illustrations are stored in museum repositories and archives that are not disclosed for generic use. Digitisation projects are currently ongoing worldwide to address this challenge, but as of now, most collections remain offline (Hedrick et al., 2020). Second, illustrations published as online digital collections can be used for research, but are often published with limited or no identifications (unique labels), which are required to study the illustrations. Most do contain captions with handwritten *historical names*, as is demonstrated in Fig. 1, but these are mostly unpublished or obsolete within today's taxonomy. Finally, the identification of an organism from a photograph or illustration, using the system of biological classification, is a complex and delicate task, even for domain experts (Austen et al., 2016).

Automated methods can significantly reduce the time and effort required by scholars to identify and classify the images. Easy access to taxonomic classifications of illustrations facilitates research into the historical abundance, range and variation of species. The current biodiversity crisis increases the importance of such historical studies as these provide a longer-term view of changes to biodiversity. In this study, we investigate: *to what extent can zero-shot learning support the classification of zoological illustrations?*

1.2. Automated classification

Photographs and illustrations of species are quite distinct. In illustrations, the background (natural habitat) is often omitted and species are depicted in the form of collages of multiple (smaller) depictions of their external and internal anatomy (e.g., bones, organs, limbs). These appear in a combination of various views (e.g., frontal, dorsal, lateral). Moreover, illustrations exist as rough pencil sketches and/or detailed colour drawings and commonly contain handwritten captions. To illustrate the differences between photographic and illustration data, three depictions and two photographs of the species *Lepas (Anatifa) anserifera Linnaeus, 1767* can be observed in Figs. 2 and 3.

The dissimilarity of the two modes demands training or fine-tuning a (pre-trained) classifier on the illustrations. However, this is a non-trivial task. For classifying zoological illustrations, only small samples from a subset of species described in modern taxonomy are available for training, and these samples are smaller for rarer species. Therefore, standard supervised classification models overfit the training data, and do not capture the totality of the problem.

Moreover, testing the model on a test-set does not guarantee its value'in the wild'. Due to various factors, there is always a divergence that affects performance: a change in distribution or differences in feature space (Wang and Deng, 2018). Illustrations, for instance, vary in use of materials, drawing style and method, and can portray zoological species unknown to the model.

1.3. Approach

Below we formulate a research approach that copes with the aforementioned challenges. To address the first challenge of handling long-tailed data with small or no samples, our approach uses a non-standard learning strategy called *zero-shot learning* (ZSL). With ZSL, it is possible to exploit data from auxiliary data sources to form semantic descriptions of classes, which can help to classify images from *unseen* classes: classes that are not observed by the classifier during training,

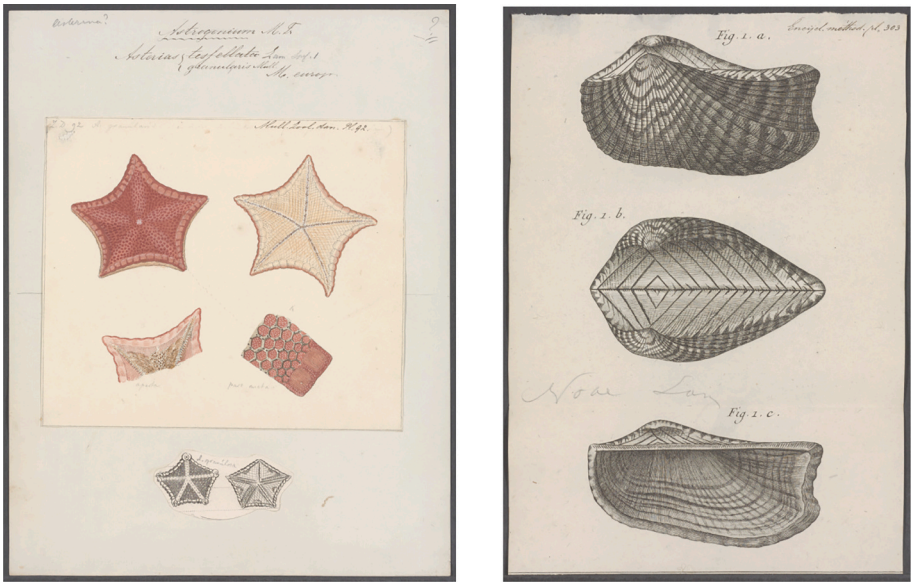


Fig. 1. Zoological illustrations from Iconographia Zoologica online https://bijzonderecollecties.uva.nl/gedeelde-content/beeldbanken/iconographia.html (best viewed in colour). Images free of known restrictions under copyright law (Public Domain Mark 1.0).

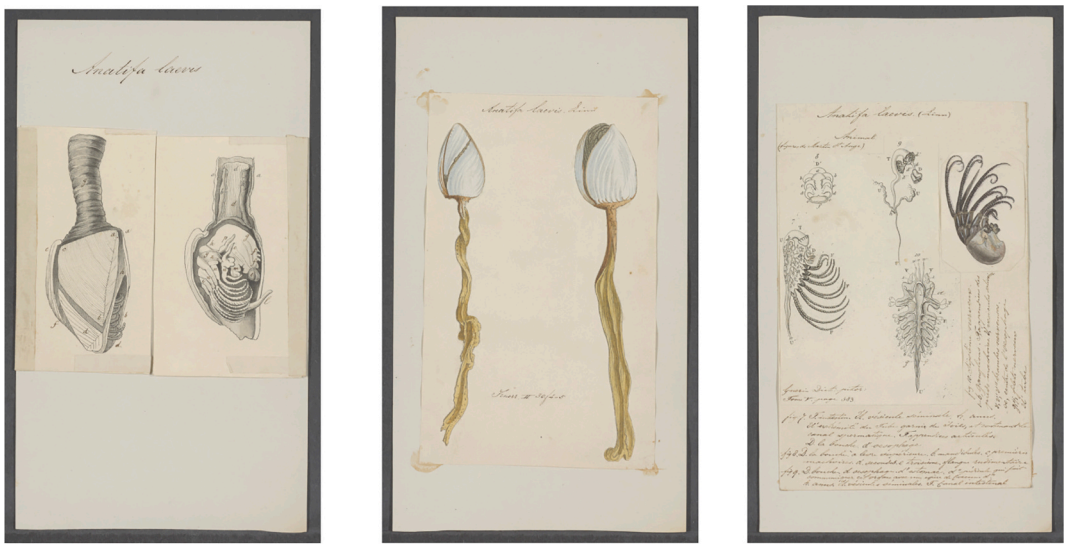


Fig. 2. Scientific illustrations from the Iconographia Zoologica 0 of *Lepas (Anatifa) anserifera Linnaeus, 1767*, with handwritten (historical) name *Anatifa laevis Bruguière, 1789* (best viewed in colour). (a) species within shell, (b) shell of species, (c) species without shell. Images free of known restrictions under copyright law (Public Domain Mark 1.0).

Fig. 3. Photographs of the species *Lepas (Anatifa) anserifera Linnaeus, 1767 (Goose Barnacle)*, taken from iNaturalist. https://www.inaturalist.org/ (best viewed in colour). (a) Observation © David R. https://www.inaturalist.org/observations/25983495 (b) Observation © mervyngreening. https://www.inaturalist.org/observations/34793791 Images are licensed under CC BY-NC 4.0.

and hereby to push the boundaries of automated recognition for a specific problem. Such a classifier is also more flexible to deal with new definitions of classes, and therefore better formulates real world conditions. This is especially useful for biological taxonomy, where the solution space is large, new class definitions can be introduced, and old ones can be revisited. To avoid overfitting, our approach additionally exploits image representations learned from another task - the recognition of zoological photographs - to extract meaningful features for our task (Oquab et al., 2014). Moreover, we use a biological taxonomy as a label hierarchy for training, and hereby have access to a larger number of labelled examples for groups higher up the label hierarchy. We evaluate our approach on the Zoological Illustration and Class Embedding (ZICE) dataset, that we introduce in this paper.

To address the second challenge, we evaluate the trained model'in the wild', on a dataset collected under different conditions. To this end, our approach uses a second independent collection of illustrations without annotations, to analyse the final species embedding model.

Our contribution is threefold:

1. We introduce the Zoological Illustration and Class Embedding dataset (ZICE) constructed from real-world data. It consists of: (i) 14,502 biological illustrations of 7973 species from the animal kingdom, with labels organised hierarchically, and (ii) class embeddings from 3 different sources - a hierarchy (taxonomy), historical texts and photographs.
2. We introduce and evaluate a zero-shot learning (ZSL) approach for fine-grained hierarchical classification. We use the prototypical networks introduced by Snell et al. in (Snell et al., 2017) and introduce: *fused prototypes* (FP), and *hierarchical prototype loss* (HPL). Our approach is evaluated on the ZICE dataset.
3. We provide a qualitative analysis of the performance of our ZSL approach in a real-world scenario on an independent verification-set: a collection of 1088 unlabelled zoological illustrations, collected during a historical biodiversity expedition (Weber, 2020).

The rest of this paper is organised as follows. In Section 2 we discuss related work on automated species classification and zero-shot learning. We discuss the data in Section 3, the methodology in Section 4, the experimental setting in Section 5 and the experiments in Section 6. We close the paper with an analysis and discussion of the results in Section 7, and our conclusions in Section 8.

2. Related work

Below, we discuss datasets related to computer vision and biodiversity, where we briefly mention recent work that leverages contextual

information for fine-grained classification, and provide a short survey of the field of zero-shot learning.

2.1. Computer vision and biodiversity

Recognising and identifying species in images is a well researched problem within the computer vision field. Most popular datasets contain classes of animals, (often birds), or plants (Beery et al., 2020b; Berg et al., 2014; Kumar et al., 2012; Lampert et al., 2009; Nilsback and Zisserman, 2006; Van Horn et al., 2015; Van Horn et al., 2018). A citizen science project called iNaturalist,[1] allows users to upload photographs of organism encounters in the wild. Since 2017, a new dataset has been published every year as part of the iNaturalist Competition FGVC6 for fine-grained image classification.[2] Computer vision models trained on such datasets are much better prepared for the automatic identification of species in the wild. Nevertheless, much variation still exists among data captured for various tasks, such as between observation data from iNaturalist, and data collected from motion-triggered camera traps.[3] Recent datasets therefore combine data captured for distinct tasks to model the variation that exists among photographs of species observations (Beery et al., 2020b).

To improve automated classification of species in images, recent work has demonstrated the usefulness of leveraging contextual data for the improvement of classification models, for instance the use of spatio-temporal data often accompanying observations to aid fine-grained classification (Beery et al., 2020a; Chu et al., 2019; Mac Aodha et al., 2019). Moreover, zero-shot learning methodologies allow researchers to leverage contextual information from multimodal sources to calculate measures of similarity between classes (Akata et al., 2015a; Sumbul et al., 2018). Such contextual information can greatly aid a model to distinguish between visually similar classes where small samples are available for training.

In addition to photographs of species, there are examples of models trained for the automated classification of plants in herbaria (Belhumeur et al., 2008). While a great deal of work is spent capturing often unclear images of species in the wild, a wealth of detailed zoological illustrations are under-utilised. Reasons could be that samples are small, many classes are under-represented, and numerous institutions have yet to start with the digitisation of their collections (Drew et al., 2017).

2.2. Zero-shot learning

While standard supervised image classification methods learn to recognise classes by observing examples of those classes during training, *zero-shot learning* (ZSL) aims to recognise classes for which no examples were observed during training, $y \in \mathcal{Y}^{ts}$, from examples of classes observed during training, $y \in \mathcal{Y}^{tr}$, using between-class feature transfer. With a training set $\mathcal{T} = \{(\mathbf{x}_1 y_1), \ldots, (\mathbf{x}_N y_N)\} \in \mathcal{Y}^{tr}$, and *embedding functions* $\varphi : \mathcal{Y} \rightarrow \widetilde{\mathcal{Y}}$ and $\theta : \mathcal{X} \rightarrow \widetilde{\mathcal{X}}$, the task is to learn a compatability function $f : \widetilde{\mathcal{X}} \rightarrow \widetilde{\mathcal{Y}}$. At test time, the function is used to classify test images from the set of unseen classes $\mathcal{Y}^{ts}$.

With θ, every image $\mathbf{x}_i \in \mathbb{R}^D$ from $\mathcal{Y}^{tr}$, is embedded in *visual feature space*, $\theta(\mathbf{x}_i) \in \mathbb{R}^M$, called an *image embedding*. Most commonly, θ is a Convolutional Neural Network (CNN). After training the CNN, the top of the network - often just the softmax layer - is removed and an image embedding function remains.

With φ, every class $y_i \in \{1, \ldots, K\}$ is mapped to a vector in *semantic embedding space*, $\varphi(y_i) \in \mathbb{R}^M$, called a *class embedding*. The semantic embedding space is either (i) created manually, through class annotations or attributes (Ferrari and Zisserman, 2007; Lampert et al., 2014), or (ii) learned from auxiliary information such as taxonomies (Barz and Denzler, 2019; Tsochantaridis et al., 2005) or texts (Harris, 1954; Mikolov et al., 2013a; Pennington et al., 2014). Attribute embeddings encode whether a certain attribute - from a set of predefined attributes - is present for a specific class. Attribute embeddings can be either binary or continuous, e.g., *{wing: 0.1, red: 0.4, tail: 0.7}* and fall within the interval [0,1]. Learned embeddings are continuous and represent similarities between classes more abstractly. Class embeddings from various sources can be used to complement one another; combining them often results in a higher accuracy (Akata et al., 2015a; Akata et al., 2015b; Sumbul et al., 2018). Combining class embeddings can be done in different ways, for instance by concatenating the class embeddings or combining compatibility scores (Akata et al., 2015a). We refer to (Akata et al., 2015a) for an extensive evaluation of class embeddings.

Most common ZSL methods learn either a linear (Akata et al., 2015a; Akata et al., 2015b; Frome et al., 2013; Romera-Paredes and Torr, 2017) or a non-linear (Socher et al., 2013; Xian et al., 2016) compatibility function between the two feature spaces. Prototypical networks (Snell et al., 2017) belong to the latter group. They learn deep visual-semantic models, such as DeViSe (Frome et al., 2013) and Cross-modal transfer (CMT) (Socher et al., 2013), in which the visual object recognition network is trained to predict the class embedding vector in semantic embedding space, which is learned from auxiliary data. While all methods achieve impressive results on small- and medium-scale datasets, the more realistic variant *generalised zero-shot learning* (GZSL), that aims to classify both seen and unseen classes, performs poorly for unseen classes (Socher et al., 2013): the model overfits to seen classes and therefore favours seen over unseen classes at test time. Hence, zero-shot learning models embedded in real world applications should include a method for dealing with this issue. For an extensive comparison of state-of-the-art of zero-shot learning and generalised zero-shot learning methods, we point to the work of Xian et al. (Xian et al., 2019). In our work we use prototypical networks for zero-shot learning because they are state-of-the-art models within the few- and zero-shot learning domain (Snell et al., 2017).

3. The data

In this section, we discuss the Zoological Illustration and Class Embedding (ZICE) dataset (see Subsection 3.1), used for training, validating and testing our zero-shot learning approach, and an independent verification-set (in Subsection 3.2) used to analyse the zero-shot learning results in a real-world scenario (in Section 7). Both datasets are published online.[4]

3.1. The ZICE dataset

The Zoological Illustration and Class Embedding (ZICE) dataset contains illustrations, from the Iconographia Zoologica online collection,[1] and class embeddings corresponding to the classes represented in the illustrations.

3.1.1. Illustrations

The Iconographia Zoologica is a 19th century collection of biological illustrations from the Artis Library of the University of Amsterdam. The collection was formed by three collectors: the well-known collector and naturalist Th. G. van Lidth de Jeude, the zoologist R.T. Maitland and the curator of the shell collection at the Amsterdam Zoo, Abraham Oltman, together with the Amsterdam society *Natura Artis Magistra*. In the 21st century, the collection was digitised and labelled with either complete binomial species names (genus and specific epithet) or corresponding genera. The full online collection contains over 26,500 pages of zoological illustrations.

[1] https://www.inaturalist.org/
[2] https://www.kaggle.com/c/inaturalist-2019-fgvc6
[3] https://github.com/microsoft/CameraTraps
[4] https://github.com/lisestork/ZICE-dataset

We have cross-referenced the illustration labels with the June 2018 backbone taxonomy (GBIF Secretariat, 2018) of the Global Biodiversity Information Facility (GBIF),[5] a central repository for biodiversity occurrence data. For 14,502 illustrations of 7973 species, labels could be cross-referenced directly with GBIF without extra domain expert curation. Matches were only accepted when the names had the status "accepted" in the GBIF taxonomy, as using labels with the status "unaccepted" or "synonym" to train a ZSL model could prove problematic. Some synonyms, for example, refer to both a plant and an animal. As a result, visual features would map to incorrect semantic representations. By the automated matching process, all classes in the ZICE dataset are organised according to a taxonomy. Fig. 4 shows twelve example illustrations.

3.1.2. Notation

A biological taxonomy can be seen as a tree datastructure, in which species are represented as leaf nodes, and parent classes represent their higher classifications based on features shared with other species. In the rest of this paper, we refer to the biological taxonomy by the term *label hierarchy*, and we refer to the various ranks (depths of the tree) by *levels*. The hierarchy consists of seven levels: kingdom, phylum, class, order, family, genus, species (genus + specific epithet). We use $\mathscr{D} = \{(\mathbf{x}_1, y_1, \boldsymbol{t}_1), \ldots, (\mathbf{x}_N, y_N, \boldsymbol{t}_N)\}$ to refer to the ZICE dataset, where each $\mathbf{x}_i \in \mathbb{R}^D$ represents a D-dimensional feature vector of an image, each $y_i \in \{1, \ldots, K\}$ represents its species label, where K thus indicates the number of leaf nodes of the label hierarchy, and $\mathbf{t}_i = [t_1, \ldots t_L]$ represents its full path of labels, one from each level and ordered from fine-grained to course-grained such that $t_i[1] = y_i$, and where L indicates the number of levels in the label hierarchy.

3.1.3. Class embeddings

To train our zero-shot learning model, we have generated class embeddings whose classes match those from the illustrations. They come from three different sources: (i) the GBIF backbone taxonomy (GBIF Secretariat, 2018), (ii) literature from the Biodiversity Heritage Library (BHL) (Gwinn and Rinaldo, 2009) and (iii) photographs from the iNaturalist 2018 challenge dataset (Van Horn et al., 2018). Information on how these embeddings are produced is given in Section 4.

3.2. The verification-set

The Committee for Natural History of the Netherlands Indies (1820–1850) was founded by King William I of the United Kingdom of the Netherlands. Their primary task was the collection of information on natural resources in the Dutch Indies. In addition, they were deployed to observe and describe the local flora and fauna (Weber, 2020). As a result, many specimens, biological illustrations and observation descriptions were brought back to the Netherlands for closer investigation, with the aim to publish results on the natural diversity of the Dutch Indies (Weber, 2020). Currently, the physical collection is stored at the *Naturalis Biodiversity Center* in Leiden. In 2008 the archival part of the collection was digitised (scanned), but due to a lack of annotation, it still remained inaccessible to biodiversity researchers. Currently, the collection serves as a use-case for the Making Sense of Illustrated Handwritten Archives Project[6] of which this work is part. We use 1088 illustrations from the collection to evaluate the model in a realistic setting. Example illustrations are presented in Fig. 5.

4. Methodology

In this section, we describe the mathematical formulation of our approach: the zero-shot learning model (ZSL) (in Subsection 4.1), image embeddings (in Subsection 4.2), class embeddings (in Subsection 4.3), our method for (i) combining class embeddings: *fused prototypes* (FP) (in Subsection 4.4), and (ii) for calculating *hierarchical prototype loss* (HPL) based on the label hierarchy (in Subsection 4.5).

4.1. Zero-shot learning model

Prototypical networks for few-shot learning, as described in (Snell et al., 2017), compute M-dimensional class representations $\mathbf{c}_k \in \mathbb{R}^M$ called *class prototypes*. They do so by embedding N_s *support points* $\{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_N, y_N)\} \in \mathscr{S}$ from N_c classes with an embedding function $f_\phi : \mathbb{R}^D \to \mathbb{R}^M$, and taking the per-class average of the resulting embedded support points, see Eq. (1). In Eq. (1), $\mathscr{S}_k$ refers to the set of support points for class k, and $\mathbf{c}_k$ refers to its calculated prototype. We further refer to the space $\mathbb{R}^M$ by the term *prototype space*.

$$\mathbf{c}_k = \frac{1}{|\mathscr{S}_k|} \sum_{(\mathbf{x}_i y_i) \in \mathscr{S}_k} f_\phi(\mathbf{x}_i) \tag{1}$$

To train the network, prototypical network loss (PNL) is calculated by mapping a set of N_q *query points*: $\{(x_1, y_1), \ldots, (x_N, y_N)\} \in \mathscr{Q}$ from the same N_c classes to prototype space. In prototype space, distances from the query points to the class prototypes are computed so that, based on a softmax over these distances, a distribution over classes is obtained. Parameters ϕ are learned by minimising the negative log-probability of the true class k via Stochastic Gradient Descent. The network is trained with mini-batches. Each mini-batch consists of N_c classes, N_q query points and N_s support points, and is called an *episode*.

For zero-shot learning, Snell et al. (Snell et al., 2017) mention that rather than embedding support points in prototype space, prototypes can be constructed by embedding auxiliary information, e.g., class embeddings in the form of attribute annotations, in prototype space. In their paper they use binary attribute vectors from the CUB-200-2011 dataset (Wah et al., 2011). They extract features from different crops of the images using a pre-trained model and map them to prototype space using a linear model with one layer. Similarly, they use a single-layer linear model to map the attributes to prototype space and prototypical training proceeds as in the few-shot setting. Rather than relying on one source (such as attributes), we rely on a combination of class embeddings from three distinct sources.

4.2. Image Embeddings

We embed images $\mathbf{x} \in \mathscr{X}$ of zoological illustrations in a lower dimensional feature space using a deep Convolutional Neural Network (CNN) $\theta(x) : \mathscr{X} \to \widetilde{\mathscr{X}}$. We will use θ to refer to the image embeddings. To make sure we don't learn features specific to our dataset (such as an illustrator's mark or a label). We transfer image representations learned from photographs (the *source* dataset) to illustrations (the *target* dataset) (Oquab et al., 2014). We use the inception V3 model (Szegedy et al., 2016), and import weights learned on the iNaturalist 2018 competition dataset.[7] For zero-shot learning, image embeddings are often generated using CNNs pre-trained on a source task (e.g., the ImageNet task (Deng et al., 2009)). The choice of model is crucial as the quality of the image embeddings has a big impact on the performance of the ZSL model. Therefore, we have chosen to use a model that was trained on a task more similar to ours. Xian et al. (Xian et al., 2019) mention that class overlap between classes from the source and target dataset leads to an unwanted positively biased result. However, our goal is not to compare between various state-of-the-art zero-shot learning methods, but rather to provide insights for training a model that is able to generalise to new data within the target domain.

[5] https://www.gbif.org/
[6] www.makingsenseproject.org
[7] https://github.com/macaodha/inat_comp_2018

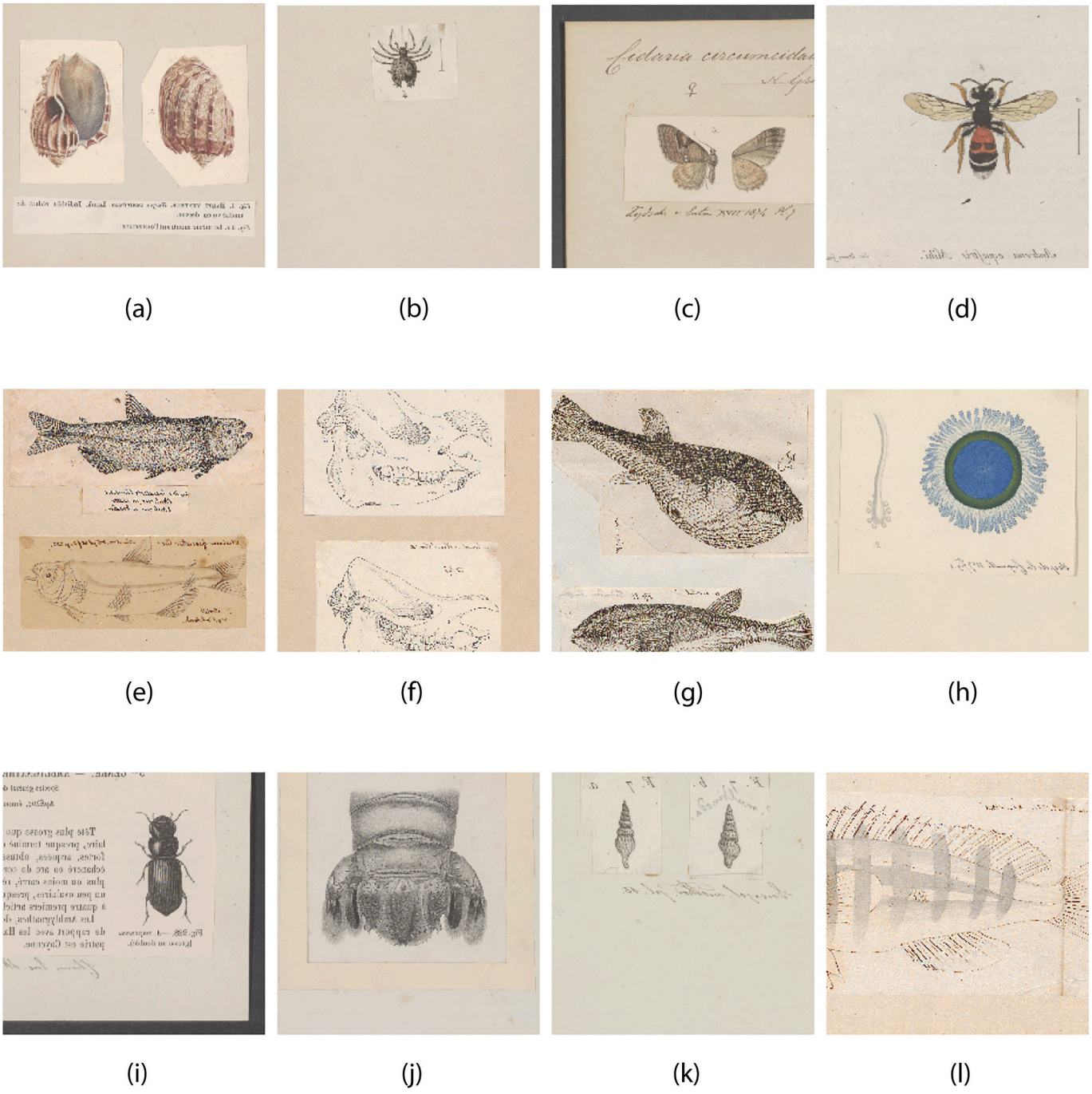


Fig. 4. Cropped example illustrations from the ZICE train-set (best viewed in colour). Image (f), depicts the skull of a *Rhinosceros unicornis* and image (j) the tail of a *Squilla hoevenii*. Images free of known restrictions under copyright law (Public Domain Mark 1.0).

4.3. Class embeddings

Below we describe details concerning the embedding functions that map classes $y_i \in \mathcal{Y}$, the set of leaf nodes from the label hierarchy, to vectors $\varphi(y_i) \in \mathbb{R}^M$ in M-dimensional semantic embedding space: $\varphi : \mathcal{Y} \rightarrow \widetilde{\mathcal{Y}}$. As each embedding comes from a different domain, all embeddings are l_2-normalised. For brevity, we use φ_k^i to refer to the class embeddings of source i for class k.

4.3.1. A hierarchy (φ^h)

Through the GBIF backbone taxonomy, we had access to the ground truth list of higher taxon labels for nearly all classes (see Table 1 for class statistics). For 53 classes, no (or an incomplete) higher classification was available. Using the deterministic algorithm from Barz et al. (Barz and Denzler, 2019), we have projected all 7920 classes onto a unit sphere of dimensionality N - where N is the number of classes. The negated dot product between classes on the sphere represents their semantic similarity. This similarity is based on the ratio of overlap between their ground truth list of higher taxon labels - nodes in the hierarchy. Part of the label hierarchy is given in Fig. 6.

4.3.2. Texts (φ^t)

To facilitate semantic search over large textual biodiversity archives, Nguyen et al. have constructed an inventory of name variants and synonyms from a large textual biodiversity corpus (BHL) (Nguyen et al.,

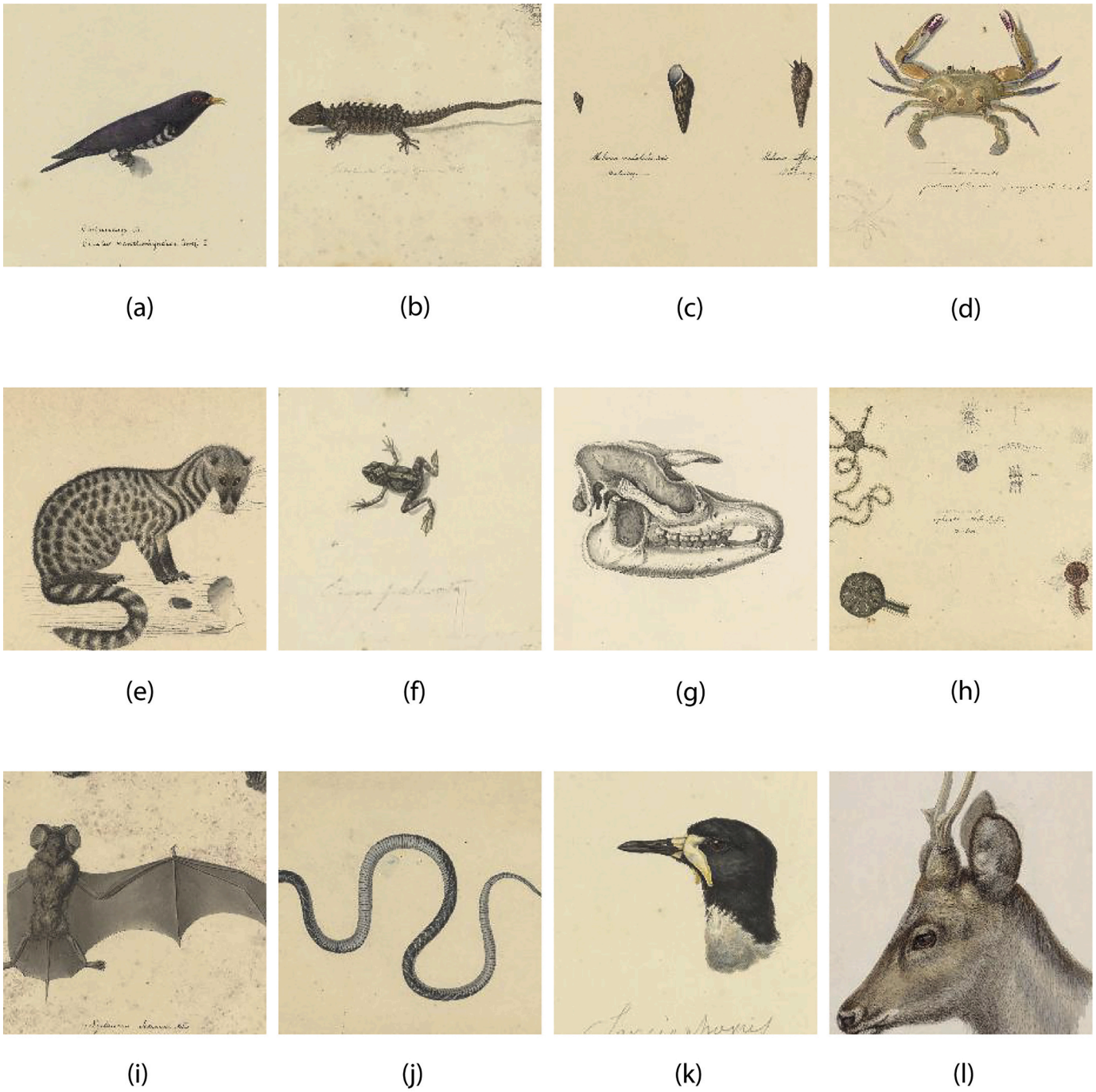


Fig. 5. Cropped example illustrations from the verification-set (best viewed in colour). Labels are unknown. Images free of known restrictions under copyright law (Public Domain Mark 1.0).

2017). For this task, they have computed word embeddings from multi-word terms -" chipping sparrows" becomes" chipping_sparrows" - mentioned in the corpus. They compared multiple methods for computing word embeddings: *continuous-bag-of-words* (CBOW) (Mikolov et al., 2013b), *count-based* (Turney and Pantel, 2010) and *Global Vectors* (GloVe) (Pennington et al., 2014). From these three, we rely on the 300 dimensional multi-word GloVe embeddings.

4.3.3. Photographs (φ^p)

Features in photographs are quite distinct from those in illustrations, but their features capture the semantic similarity of the different classes they represent in a similar way. Hence, we have extracted 2048 dimensional features from the iNaturalist 2018 dataset photographs, using the inception V3 model trained on the corresponding dataset (previously mentioned in Section 4.2).

4.4. Combining class Embeddings

Below we describe two methods for generating singular class prototypes for prototypical learning (see Section 4.1) from three distinct embeddings, each with a different dimensionality.

4.4.1. Concatenated embeddings (CE)

One method that is often employed to combine the different

Table 1
Dataset statistics per super-class (phylum) total number of leaf node classes $\mathscr{Y}$ and instances N, the number of leaf node classes per split $\mathscr{Y}^s, s \in \{tr, v, ts\}$, the number of instances per split N^s, and the number of leaf node classes per embedding $\mathscr{Y}^{\phi^e}, e \in \{h, t, p\}$.

Super-class (phylum)	$\mathscr{Y}^{tot}$	N^{tot}	$\mathscr{Y}^{tr}$	$\mathscr{Y}^{v}$	$\mathscr{Y}^{ts}$	N^{tr}	N^{v}	N^{ts}	$\mathscr{Y}^{\phi^h}$	$\mathscr{Y}^{\phi^t}$	$\mathscr{Y}^{\phi^p}$
Arthropoda	2977	3740	620	1106	1251	1383	1112	1245	2977	218	14
Chordata	2903	7358	1281	870	752	5736	878	744	2901	2050	475
Mollusca	1423	2384	488	464	471	1449	464	471	1385	475	40
Cnidaria	179	299	58	47	74	178	48	73	179	88	5
Echinodermata	111	180	36	33	42	105	33	42	111	62	10
Annelida	109	171	32	44	33	94	44	33	106	61	3
Porifera	59	79	17	17	25	37	17	25	59	11	–
Platyhelminthes	56	75	9	38	9	28	38	9	55	9	–
Bryozoa	45	67	10	12	23	32	12	23	45	23	–
Brachiopoda	37	38	1	23	13	2	23	13	37	2	–
Nematoda	18	24	4	9	5	10	9	5	18	8	–
Rotifera	17	20	2	7	8	5	7	8	17	12	–
Ctenophora	14	33	5	2	7	24	3	6	14	6	–
Nemertea	6	8	2	3	1	4	3	1	4	4	–
Sipuncula	5	6	1	3	1	2	3	1	5	–	–
Acanthocephala	4	5	1	1	2	2	1	2	4	2	–
Nematomorpha	2	6	1	1	–	5	1	–	2	1	–
Onychophora	2	2	–	2	–	–	2	–	0	2	–
Cephalorhyncha	1	1	–	1	–	–	1	–	0	1	–
Chaetognatha	1	2	1	–	–	2	–	–	1	1	–
Entoprocta	1	1	–	1	–	–	1	–	0	1	–
Animalia	3	3	–	2	1	–	2	1	0	3	–
Total	7973	14,502	2569	2684	2717	9098	2702	2702	7920	3040	547

embeddings is concatenation, in which the dimensions of each class embedding (from distinct sources) are concatenated together. This results in one sparse matrix with a large dimensionality. Similarly to Snell et al. (Snell et al., 2017), we learn a one-layer linear model on top of the concatenated class embeddings φ and on top of the the image embeddings θ, mapping both to prototype space.

4.4.2. Fused prototypes (FP)

We implement *fused prototypes*, see Fig. 7. Essentially, fused prototypes fuse prototypes from a variable number of multimodal sources into a single prototype per class. We derive our implementation from the prototypical few-shot learning approach. Instead of using support points $s \in \mathscr{S}$, we use $\varphi^i \in \Phi$, the set of class embeddings from distinct sources $\{\varphi^1, \ldots, \varphi^N\}$. A simple one-layer linear model is learned on top of the feature space of each of the distinct φ^i's as well as the image embeddings θ, mapping both to prototype space. In prototype space, the embedded φ^i's are fused together, similarly to the way support points are fused to form class prototypes for few-shot learning, see Eq. (2).

$$\mathbf{c}_k = \frac{1}{|\Phi|} \sum_{\left(\varphi_k^i, y_k\right) \in \Phi} f_{\phi^i}\left(\varphi_k^i\right) \tag{2}$$

In that equation, $\mathbf{c}_k$ refers to the class prototype for class k, and f_{ϕ^i} refers to the linear model that maps the individual class embeddings from φ^i to prototype space. We hypothesise that fused prototypes will perform better than concatenated embeddings, as the latter introduce one large sparse input space whereas fused prototypes are optimised from multiple dense input spaces.

4.5. Hierarchical prototype loss

Hierarchical prototype loss (HPL) extends prototypical network loss (PNL), and is defined as the sum of the losses for each level of the label hierarchy (see Fig. 6). The loss for a specific level l is calculated by first computing temporary parent-class prototypes $\mathbf{p}_k \in \mathbb{R}^M$ for that level from the set of class prototypes $\mathscr{C} = \{(\mathbf{c}_1 y_1, t_1), \ldots, (\mathbf{c}_N y_N, t_N)\}$, see Fig. 7 and Eq. (3). In the Equation, $\mathscr{C}_k$ refers to the subset of $\mathscr{C}$ containing all prototypes $(\mathbf{c}_i, y_i, t_i)$ where $t_i[l] = k$. As described in Section 4.1, distances of the query points to the temporary parent-class prototypes are then computed and the loss is calculated over these distances. The HPL is calculated by summing the losses for all L levels.

$$\mathbf{p}_k = \frac{1}{|\mathscr{C}_k|} \sum_{(\mathbf{c}_i y_i, t_i) \in \mathscr{C}_k} \mathbf{c}_i \tag{3}$$

By implementing HPL, we take a multi-granularity approach: we enforce a clearer separation of classes not only for the finest grain, but also for coarser taxonomic groups. As more labels are available for each level higher up in the label hierarchy, this intuitively supports the discovery of more robust features for the classification of coarser classes.

5. Experimental setting

In this section we discuss details regarding the settings of the experiment: the dataset splits (in Subsection 5.1), data augmentation (in Subsection 5.2), and evaluation criteria (in Subsection 5.3).

5.1. Dataset splits

As recommended by (Xian et al., 2019), we split the classes $\mathscr{Y}$ for training and evaluation based on the number of instances each of them contain. Since our dataset contains so few instances per class, ($n_k \in [1,283]$, μ: 1.79, σ: 3.93). We have used all classes with $n \geq 2$ per class for the training set $\mathscr{Y}^{tr}$. Two examples per class is not sufficient to learn a good class representation, but the features of these illustrations are useful for between super-class feature sharing. Moreover, we exploit them for learning representations of classes on a higher taxonomic level, since a larger number of instances are available higher up the label hierarchy. All remaining classes with $n = 1$ were equally distributed over the validation set $\mathscr{Y}^{v}$, and the test set $\mathscr{Y}^{ts}$. Table 1 shows dataset statistics per super-class. Since not all of the classes were represented in each source (GBIF, BHL and iNaturalist), each embedding (φ^h, φ^t, and φ^p respectively) represents a subset of $\mathscr{Y}$. However, together they span the totality of classes $\mathscr{Y}$. The super-class *Animalia* is used for classes that are not assigned to a phylum.

5.2. Data augmentation

For training, we used image embeddings extracted from augmented versions of all images, in order to increase the ability of the classifier to generalise the classification with respect to the data. Before cropping all

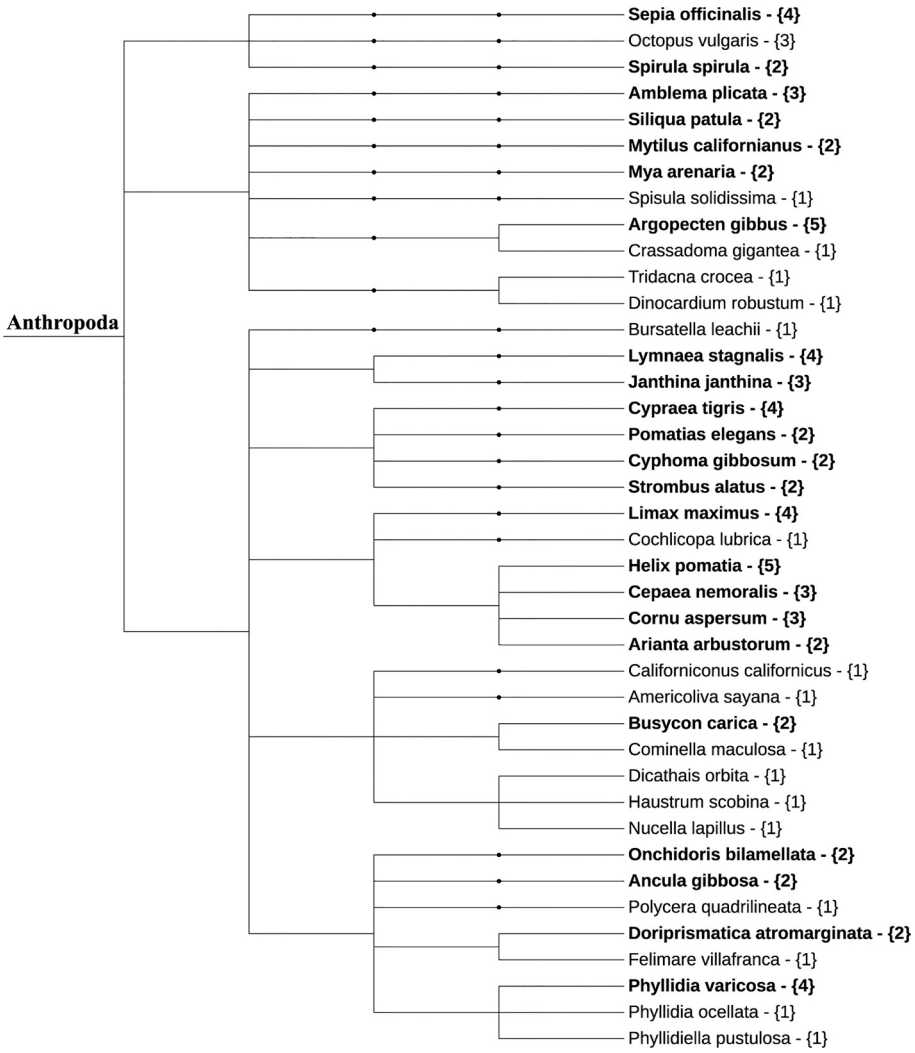


Fig. 6. A subset of $\mathscr{Y}$ from the ZICE dataset, covering the phylum *Anthropoda*, with the corresponding label hierarchy (from left to right: phylum to species). **Bold** names indicate classes from $\mathscr{Y}^{tr}$, and numbers indicate number of instances within that class.

images, the largest side of each image was first resized to 300. During resizing, we kept the aspect ratio identical to the original image. 2048-dimensional features were extracted by applying the pre-trained Inception V3 model to crops (middle, upper left, upper right, lower left and lower right) of each resized original illustration and its horizontally flipped version. Crops containing only white space or text were manually discarded.

5.3. Evaluation criteria

In our experimental ZSL results (Subsection 6.2) we report two accuracy metrics: top-*k* accuracy and hierarchical accuracy@*k*.

5.3.1. Top-k accuracy

Flat top-1 accuracy does not always sufficiently portray the classifier's capabilities. When the solution space is large, it is valuable for domain experts to obtain top-*k* predictions, as exemplified later in Fig. 9. We therefore report top-*k* accuracy, $k \in \{1,2,5,10\}$. This metric is computed by the percentage of images for which the correct label is among the top k predictions.

5.3.2. Hierarchical accuracy@k

For our task, classifying an illustration of a *Boiga nigriceps* as a *Boiga dendrophila* - both tree snakes - is less problematic than classifying it as a *Procyon lotor*, a common raccoon. In the former case, the classifier has learnt important coarse features specific to tree snakes, and has provided researchers with a partially incorrect, but valuable classification nonetheless. For each illustration, we would therefore like to shed light on the accuracy of the entire label path from the label hierarchy. Hence, we additionally report *hierarchical accuracy*. Hierarchical @k precision is sometimes used as a metric for hierarchical datasets (Frome et al., 2013). We report a new metric that we deem more informative in our context: *average per-level accuracy*, or *hierarchical accuracy*. It is computed by calculating the accuracy for each level in the label hierarchy and averaging over these, see formula 4. In formula 4, L refers to the number of levels for which we have labels and l to a specific level l.

$$Hierarchical\ acc = \sum_{l=1}^{L} \frac{n \text{ correct preds in } l}{n \text{ samples in } l} \tag{4}$$

Additionally, we report accuracies for labels k levels up the label hierarchy, where $k \in \{1,2,3\}$, thus referring to the accuracy for labels one, two and three levels up the label hierarchy respectively.

6. Experimental results

The following section is divided as follows: first we evaluate the image embeddings (**Task 1**) in a supervised classification setting (Subsection 6.1), after which we evaluate each of the elements of our zero-

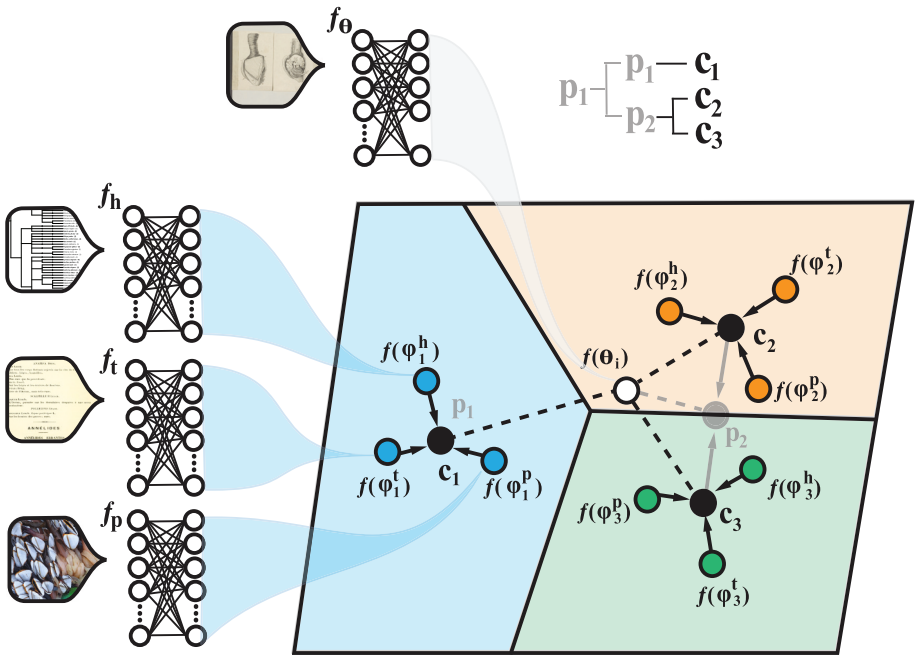


Fig. 7. Fused prototypes (FP) (best viewed in colour). Figure derived from (Snell et al., 2017). Features from φ^i (here i is replaced by: a hierarchy (h), texts (t), and photographs (p)) are mapped into prototype space using separate one-layer linear models f_{ϕ^i}, and fused into one prototype per class $\mathbf{c}_k$. To illustrate hierarchical prototype loss (HPL), example temporary parent-class prototypes $\mathbf{p}_k$ are depicted in transparent grey.

shot learning approach (Subsection 6.2): the class embeddings (**Task 2**), combining class embeddings (**Task 3**), hierarchical prototypical loss (**Task 4**), and an analysis of the final network, which incorporates results from Task 2–4 (**Task 5**).

6.1. Supervised classification and visualisation

For Task 1, we selected image embeddings from the set of species that is disjoint from the set of species represented in the iNaturalist 2018 dataset (on which the embedding function was trained), so as to obtain a deeper insight into the ability of the embedding function to find generic features. From this selection, we again selected a subset for classification and visualisation purposes: the 12 most populated classes from the *family* level (two levels up the label hierarchy).

We show per-class, micro, macro and weighted average precision and recall results for a Support Vector Machine (SVM) trained on the subset, see Table 2. Additional to family labels (Table 2, 2 column), we show higher-taxon labels from the class level (Table 2, 1 column). The weighted average alters the macro metric to account for label imbalance. The support column indicates the number of actual occurrences of that class in the given subset.

The SVM was trained using a stratified 80%–20% split for the train and test-set, respectively. Note that the classification results serve to provide an insight into the quality of the features rather than the difficulty of our task. For visualisation, we show a t-Distributed Stochastic Neighbor Embedding (t-SNE) (Maaten and Hinton, 2008) visualisation of the subset with family labels (see Fig. 8(b)). Also here, we present higher-taxon labels from the class level (see Fig. 8(a)).

Looking at Fig. 8, we see that same-class image embeddings are visibly clustered. However, classes within certain taxon groups overlap, for instance, families within the class *Mammalia*, see the classes of Fig. 8 (b) that are colored brown in Fig. 8(a). This effect is reflected in Table 2 (see **bold** text): the image embeddings from only one of four families subsumed under the class *Mammalia* can be classified correctly (*Canidae*, with 100% recall). From the classifications and the precision value (48%) we find that image embeddings from other classes subsumed under the class Mammalia are also classified as Canidae, and thus a large part of the brown cluster from Fig. 8 is classified as the family Canidae (dog-like carnivores).

The results of Task 1 show us that the features learned from the iNaturalist 2018 task are not sufficiently specific to properly classify all fine-grained classes in our task well. Therefore, further improving the image embeddings would improve zero-shot learning results, although the inter-class variation of species within certain taxon groups can be quite small. Some species within the order *Coleoptera* (beatles), for instance, can only be accurately identified after a close inspection of their genitalia (Choate, 1999). Visualisation of the features can give an indication up to which grain the features within specific taxon groups are sufficiently informative for proper classification.

Table 2
Classification precision, recall and f1 results for **Task 1** in % (rounded off to whole integers) for a Support Vector Machine (SVM) trained on the image embeddings belonging to 12 families (also visualised in Fig. 8(b)). Support indicates the number of actual occurrences of that class in the given subset. The top-1 per-class average accuracy is 43.58%.

Class	Family	Precision	Recall	f1	Support
Mammalia	**Bovidae**	**0**	**0**	**0**	**19**
Mammalia	**Canidae**	**48**	**100**	**65**	**33**
Insecta	Carabidae	44	74	56	27
Insecta	Cerambycidae	56	85	68	26
Mammalia	**Cercopithecidae**	**0**	**0**	**0**	**9**
Gastropoda	Conidae	87	98	92	41
Insecta	Curculionidae	0	0	0	14
Mammalia	**Equidae**	**0**	**0**	**0**	**12**
Insecta	Melolonthinae	100	22	36	9
Gastropoda	Muricidae	67	55	60	11
Insecta	Staphylinidae	0	0	0	10
Bivalvia	Veneridae	82	90	86	10
	micro avg	60	60	60	221
	macro avg	40	44	38	221
	weighted avg	46	60	50	221

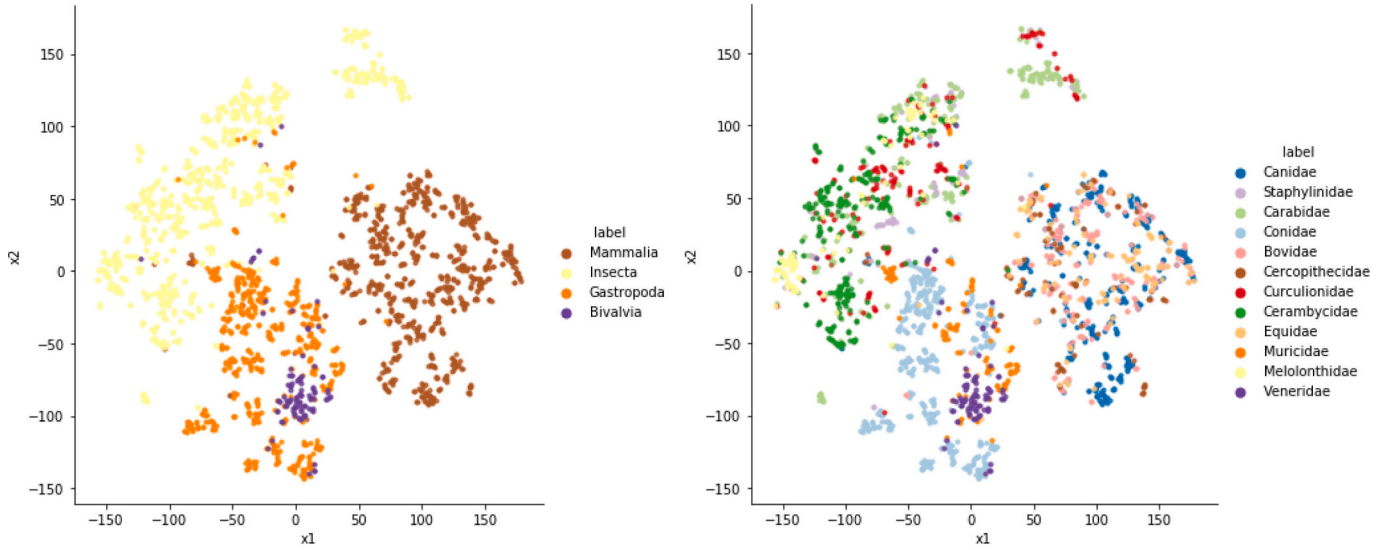


Fig. 8. t-SNE plots showing image embeddings of images from the ZICE dataset (should be viewed in colour). Plot (a) shows class level labels and (b) family level labels. Family labels come from a selection of 12 families of which the binomial name was not present in the iNaturalist 2018 dataset. The t-SNE algorithm was run for 5000 iterations with perplexity 20.

Table 3
Zero-shot learning (ZSL) classification results in % for **Task 2, 3, 4** and **5**. The 50-way classification accuracy for the final model was 35.53%, calculated by averaging results over 6000 randomly drawn episodes.

Task	Method	φ^h	φ^t	φ^p	Top-k acc $\mathscr{Y}^{ts}$				Hierarchical acc@k $\mathscr{Y}^{ts}$			
					1	*2*	*5*	*10*	*1*	*2*	*3*	*avg*
Task 2	N/A	✓	×	×	**2.29**	**4.12**	**8.9**	**15.34**	**5.93**	13.23	43.74	36.38
		×	✓	×	0.41	0.66	1.14	1.72	0.72	1.22	7.33	12.53
		×	×	✓	0.55	0.85	1.47	2.15	1.03	2.81	15.29	18.26
	FP	✓	✓	×	2.13	3.89	**8.79**	**15.11**	5.51	13.56	43.21	35.96
		✓	×	✓	**2.50**	**4.26**	**8.91**	**15.26**	**6.05**	**14.24**	**45.69**	**36.85**
		×	✓	✓	0.53	0.84	1.45	2.06	1.04	2.02	9.41	13.50
		✓	✓	✓	**2.42**	**4.29**	**9.10**	**15.37**	**5.98**	**14.22**	**45.09**	**36.70**
Task 3	CE (baseline)	✓	✓	✓	2.09	**4.05**	**8.96**	**15.54**	5.45	13.42	**44.76**	36.41
	FP	✓	✓	✓	**2.42**	**4.29**	**9.10**	**15.37**	**5.98**	**14.23**	**45.09**	**36.70**
Task 4	FP + PNL	✓	✓	✓	**2.42**	**4.29**	**9.10**	**15.37**	**5.98**	14.23	45.09	36.70
	FP + HPL	✓	✓	✓	2.12	3.88	**8.88**	**15.03**	**6.23**	**15.71**	**51.10**	**39.35**
Task 5	Final model	✓	×	✓	2.77	4.74	9.64	16.02	6.94	16.65	50.71	39.67
	Majority guess	–	–	–	0.04	0.07	0.19	0.37	2.85	3.26	21.87	18.66

6.2. *Fine-grained zero-shot learning*

All prototypical networks were trained using the Adam optimisation algorithm from pytorch.[8] Episodes for training were comprised of $N_c = 50$, $N_q = 1$ and $N_s = 0$, similar to a balanced mini-batch of size 50. The validation loss was monitored during training and if, for 10 iterations, the loss did not decrease, the learning rate was decreased with a factor of 0.5. We tuned hyper-parameters using hyper-parameter optimisation - tree-structured parzen estimators - and ended up with a learning rate of 10^{-4} and a weight decay of 10^{-5}. Early stopping on the validation loss was used to determine the optimal number of epochs for training. For each model, five different networks were trained. As a statistical test for comparing classifiers we used the McNemar test (Dietterich, 1998) for each classifier pair for all predictions of 5 runs accumulated. It is a test that works well for testing statistical significance when dealing with paired nominal data for comparing classifiers trained, validated and tested multiple times on the same splits of a dataset. **Bold** numbers indicate statistical superiority over other values within that column and cell (which separates tasks). A final model was trained, again 5 times, with the configuration that we found to work best. The last row of Table 3 indicates accuracy values for the majority guess, where the model simply always predicts the majority class.

6.3. *Evaluation (Task 2, 3 and 4)*

First, Table 3 presents results for Task 2, which show the performance of the networks trained, validated and tested with embeddings from each unique source separately, and additionally each combination of the three distinct embeddings . In order for the results to be comparable between all combinations, we used the totality of $\mathscr{Y}$ to train, validate and test the networks, despite the fact that each φ^i spans a subset of classes from $\mathscr{Y}$ (see the last row of Table 1). In case a class k was not represented in φ^i, the dimensions for φ^i_k were set to zero. In this context, the results inform us, first and foremost, about the contribution of each embedding to the overall accuracy (Table 3, Task 2, last row). We discuss each embedding separately.

φ^h is the most complete and informative embedding. φ^t spans many classes (3040 out of 7973), but appears less informative. The prototypical network trained with φ^t performs better than the majority guess for

[8] https://pytorch.org/docs/stable/optim.html

the top-k acc metric, but φ^t seems to harm the learning ability of the network when used in combination with other embeddings. This could be due to a myriad of factors. We believe the two most likely factors are that (i) the embedding is better suited for finding synonyms between taxon terms - as similar species are described similarly, and, (ii) that some names in the Biodiversity Heritage Library (Gwinn and Rinaldo, 2009) are ambiguous: referring to one species in the historical texts, while they refer to another in modern taxonomy. Particularly, any historical unpublished name could have been published today as a different species. Matching them with sources from a modern taxonomy could therefore be problematic. Finally, the network trained with φ^p shows improvement over the majority guess, and φ^p complements φ^h, as the network trained with $\{\varphi^h, \varphi^p\}$ improves over the accuracy of the model trained with just $\{\varphi^h$ (see Table 3, Task 2, row 1 and 5), specifically the hierarchical acc@2 (13.23% to 14.24%) and @3 (43.74% to 45.69%). We hypothesise that if we increase the number of instances and fine-grained classes used to generate φ^p, results could be improved further.

Second, Table 3 presents results for Task 3: combining class embeddings. CE represents the baseline model: it is comparable to the method used by Snell et al. (Snell et al., 2017) for zero-shot learning. Results for Task 3 show us that by using our fused prototypes (FP) formulation, we can increase the top-1 accuracy from 2.09% to 2.42% (see Table 3, Task 3). Such an increase is non-trivial. As the test-set contains an instance per class, with a total of 2702 classes (on the finest grain), an increase of 0.33% for the top-1 accuracy equals the capability of the classifier to correctly classify illustrations from an *additional* 9 unseen classes from different parts of the biological taxonomy. Fused prototypes also induce a higher hierarchical accuracy @1 and @2 (from 5.45% to 5.98% and 13.42% to 14.23%, respectively). When class embeddings from additional (informative) sources are used, we anticipate that this effect which we discuss in Section 4.4 will become more evident: the value of using fused prototypes over concatenated embeddings will increase.

Third, Table 3 gives results for Task 4, which show that using hierarchical prototype loss (HPL) improves the average hierarchical accuracy significantly - from 36.70% to 39.35%. However, a decrease is measured for the top-1 and top-2 accuracy: from 2.42% to 2.12% and 4.29% to 3.88% respectively. This effect demonstrates intra super-class variation of taxon groups, as it appears that learning better coarser features slightly complicates the classification of some fine-grained taxon groups.

Table 5
Generalised zero-shot learning (GZSL) classification results in % for final model.

Method	top-k acc $\mathcal{Y}^{ts}$				Hier. acc@k $\mathcal{Y}^{ts}$		
	1	*2*	*5*	*10*	*1*	*2*	*avg*
GZSL	0.04	0.21	1.24	3.25	4.47	16.03	38.19
M. guess	0.01	0.03	0.06	0.13	2.85	3.26	18.66

6.4. Final results (Task 5)

A final model was trained 5 times using the best configuration - $\{\varphi^t, \varphi^p\}$, FP and HPL. Although implementing HPL decreases the top-1 and top-2 accuracy, a substantial increase of the average hierarchical accuracy was measured. We therefore chose to implement it in the final model.

Table 3 (Task 5) shows per-network averaged results for the final model on the test-set, and Table 4 gives results for the final model's best network, detailed per super-class. Table 4 serves to provide a deeper insight into the trained network. Evidently, illustrations from some super-classes were not recognised at all due to their limited contribution to the training of the network - visible from the column avg. N_{ts} - as most feature sharing occurs within super-classes. For reason of comparison we add the results of the leaf node level (species).

On top of these results, Table 5 details results for generalised zero-shot learning (GZSL). The top-k accuracies for GZSL are poor: during classification, a network trained for ZSL tends to favour seen classes over unseen classes (Socher et al., 2013). Logically, GZSL does not affect the average hierarchical accuracy by much, as seen and unseen classes share parent-classes (see Fig. 9).

Finally, we present and discuss four example images from the test-set with their top-5 predictions (and corresponding confidence values), see Fig. 9.

Image (a) and (b) have good top-5 predictions: the top-1 prediction of image (a) is incorrect (the classifier is most confident about the label *Brachirus macrolepis*, while the correct label is *Brachirus panoides*), but the top-1 prediction is correct up to the fine-grained genus level: *Brachirus*. Moreover, the top-3 predictions are all correct up to the genus level. For image (b), the top-1 prediction is correct, and the remaining predictions are from the same correct order.

The third image (c) has poor predictions, as (i) the correct label is not among the top 5 predictions, and (ii) almost all predictions are from a different phylum. Interestingly, however, the top-2 predictions (the *Bittium reticulatum* and *Cyclura cornuta*) have something in common with

Table 4
Zero-shot learning (ZSL) classification results in % for **Task 5** on the test-set per super-class (phylum).

			Top-k acc $\mathcal{Y}^{ts}$				Hierarchical acc@k $\mathcal{Y}^{ts}$			
Super-class (phylum)	N_{tr}	N_{ts}	*1*	*2*	*5*	*10*	*1*	*2*	*3*	*avg*
Chordata	5736	744	4.7	7.39	14.65	24.06	14.65	53.36	81.05	50.22
Mollusca	1449	471	3.4	6.16	11.89	20.59	29.3	47.56	73.25	47.77
Arthropoda	1383	1245	1.61	2.97	6.59	10.6	15.74	60.88	80.0	50.1
Cnidaria	178	73	8.22	9.59	16.44	30.14	19.18	31.51	41.1	29.86
Echinodermata	105	42	4.76	7.14	9.52	21.43	9.52	11.9	33.33	19.05
Annelida	94	33	0.0	0.0	0.0	0.0	0.0	0.0	3.03	1.21
Porifera	37	25	0.0	8.0	8.0	16.0	4.0	8.0	44.0	20.0
Bryozoa	32	23	0.0	0.0	0.0	8.7	0.0	4.35	4.35	3.48
Platyhelminthes	28	9	0.0	0.0	0.0	0.0	0.0	0.0	11.11	4.44
Ctenophora	24	6	0.0	0.0	33.33	33.33	0.0	0.0	0.0	3.33
Nematoda	10	5	20.0	20.0	40.0	40.0	20.0	40.0	40.0	32.0
Rotifera	5	8	0.0	0.0	0.0	12.5	0.0	0.0	0.0	0.0
Nemertea	4	1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Sipuncula	2	1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Brachiopoda	2	13	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Acanthocephala	2	2	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Animalia	0	1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Per super-class average	534.76	158.94	2.51	3.6	8.26	12.79	6.61	15.15	24.19	15.38
Per leaf node (species)	9098	2702	2.96	4.96	9.96	16.65	7.11	17.10	52.26	40.05

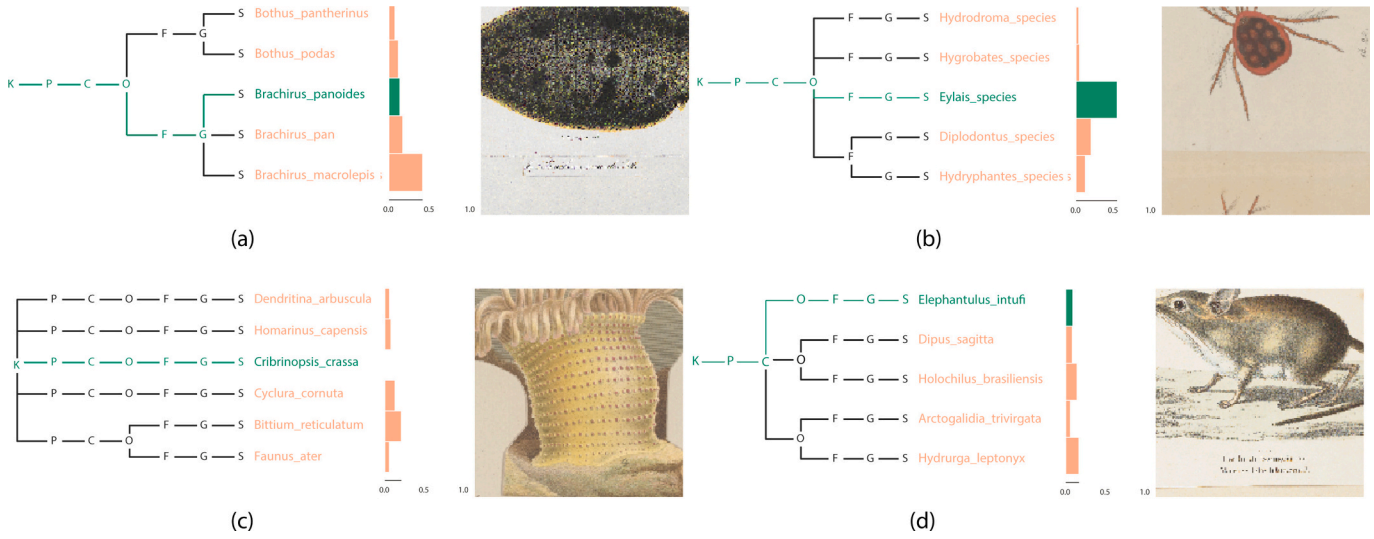


Fig. 9. Top 5 predicted classes (on the species level) and their confidence values for four example test images (best viewed in colour). Labels are organised hierarchically (K: kingdom to S: species) to show the diversity of predictions and how close - in the label hierarchy - the classifier is to the real label. Image (c) shows six predictions, as the correct label was not among the top 5 predictions. A dark green path, label and confidence bar denotes the correct label. Orange confidence bars indicate incorrect predictions. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

the correct species (*Cribrinopsis crassa*): they share its most salient feature - their skin is covered with small tubercles.

Lastly, for the fourth image (d) the correct label (*Elephantulus intufi*) belongs to the order *Macroscelidea* (Elephant shrew), and the other predictions belong to the orders (from top to bottom): *Rodentia* (Rodents) and *Carnivora* (Carnivores). The two predictions from the *Rodentia* order are two different *mice* species (*Dipus sagitta* and *Holochilus brasiliensis*. Elephant shrew visually resemble mice. Interestingly, the most salient feature that would allow a classifier to distinguish between a mouse and an elephant shrew, is cut off from the illustration: its long

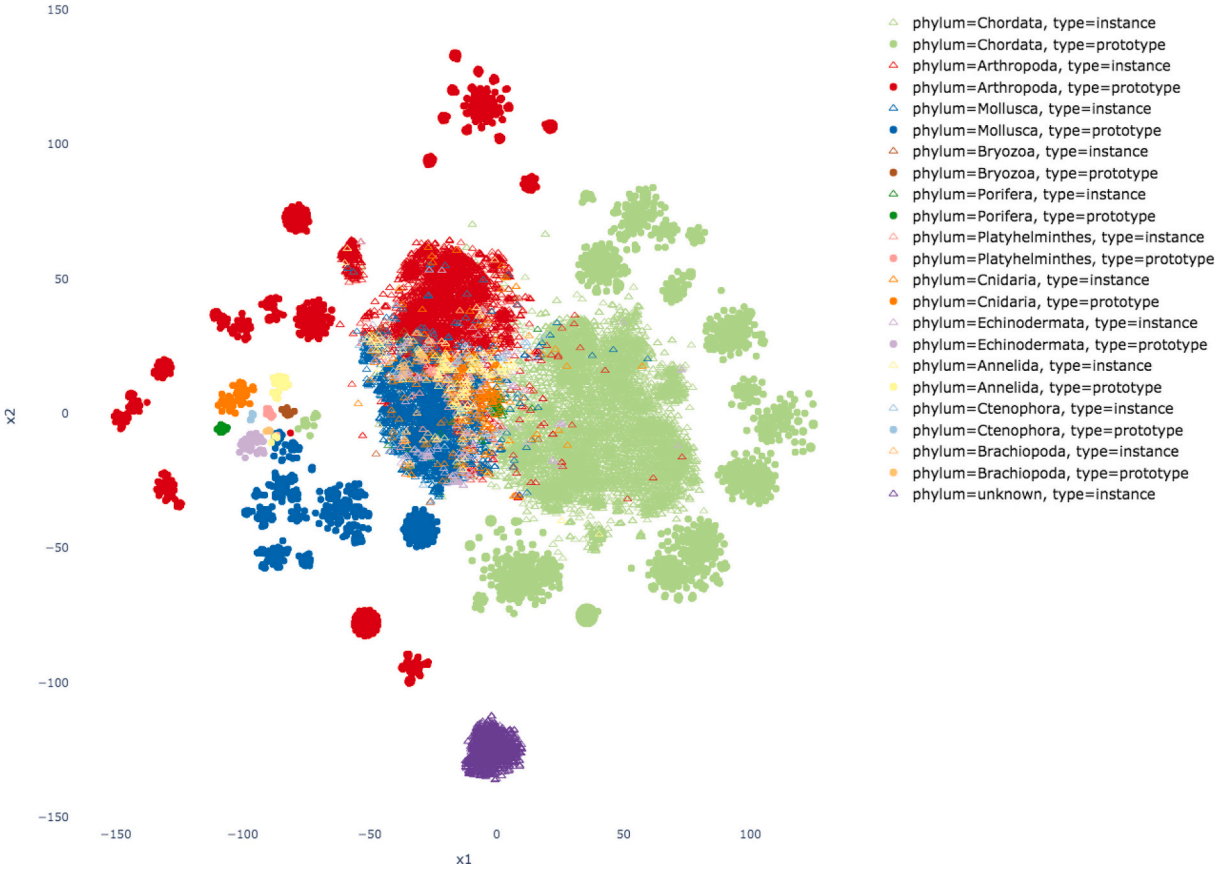


Fig. 10. A t-SNE plot showing all prototypes (closed circles) and instances (open triangles), from the 12 most populated phyla, embedded by the final prototypical network (should be viewed in colour). Instances from the verification set (bottom cluster) are indicated by the label 'unknown'. Note that t-SNE does not accurately preserve distances between clusters. The t-SNE algorithm was run for 5000 iterations with perplexity 100.

trunk-like nose, which resembles an elephant's trunk. It is therefore good to consider that cropping the image at its center in a standardised way can cause the loss of information that is vital for proper classification.

7. Analysis and discussion

Standard supervised classification offers limited solutions to deal with the full scope of the problem presented above. ZSL models are better suited to deal with limited data (small samples for only a subset of classes from the domain). For instance, Table 5 shows that 20 *Anthropod* species could be correctly classified without any training examples, from their similarity to 620 other seen *Anthropod* species. We note that this shows an important gain: the labelling of these illustrations by domain experts is costly, and does not necessarily guarantee high-quality annotations, due to the complex nature of species classification (Austen et al., 2016). Especially prototypes optimised according to the label hierarchy can be exploited in an expert support system to guide experts in the identification process.

In practice, it can be a real challenge to transfer results to real-world scenarios. We provide two telling examples. First, Table 5 shows us that with GZSL, seen classes are favoured over unseen classes during classification. In real-world applications, methods are required that deal with this issue. If not, a classifier will often prefer classes from $\mathcal{Y}^{tr}$ over $\mathcal{Y}^{ts}$ for classification. Second, using a trained network in real-world applications can prove problematic due to a domain shift between datasets. Our verification-set, that we have presented in Section 3.2, serves to illustrate this issue. When using the final species embedding model for classification of the verification-set, all instances are classified as species of *Anthropods*, although it contains illustrations from a variety of phyla (among which Chordates and Annelids, see Fig. 5). The t-SNE visualisation, see Fig. 10, allows us to hypothesise about the results. The visualisation shows instances from the verification-set (depicted as purple triangles, see bottom cluster), as well as instances and prototypes from the ZICE dataset (all other open triangles and closed circles respectively), all embedded by the species embedding model. The species embedding model appears to have mapped instances from the verification-set to a different manifold than those from the ZICE dataset. Consequently, instances from the verification-set manifold are classified as *Anthropods*, as its prototypes are closest (see the red prototype clusters in Fig. 10). We hypothesise that both datasets must come from a distinct marginal probability distribution. Most likely, this domain shift is the result of differences in paper types, sketching techniques and materials.

Overcoming the aforementioned issues is key, but we argue that ZSL and hierarchical learning methods (methods that exploit the label hierarchy) are fundamental for problem domains such as the one described here: where labelling of images is expensive, but where, at the same time, auxiliary data sources contain a wealth of domain knowledge maintained by a community of experts.

8. Conclusions

In this paper we have analysed the problem of classifying species in zoological illustrations. For this purpose, we have introduced a dataset, with many classes and few samples, and an independent (unlabelled) verification-set, both representative of the problem domain.

From the experimental results, we conclude that auxiliary data sources have allowed us to push the boundaries of automated recognition for this specific problem: illustrations from 80 classes, that contained zero example instances for training, could be classified correctly. We furthermore conclude that our model improves over the baseline classifier. Compared with the baseline, our FP implementation allowed us to classify instances from an additional 9 unseen fine-grained classes. Moreover, implementing HPL increased the average hierarchical accuracy substantially (from 36.41% to 39.35%). Finally, from the results of the analysis of the verification set in Section 7, we show the complexity of our task. Aside from the depicted illustrations, there are other differences between the digital images that impact the predictive capabilities of the model. The illustrators' technique, the physical drawing materials and the chosen perspectives change significantly between illustrators. In order for our zero-shot learning model to function well in an application, domain adaptation methods should be employed to align domain marginal probability distributions (Wang and Deng, 2018) between datasets, and therefore make the model illustrator-invariant.

Coming back to our main problem description, we conclude that computational methods support the development of species embedding models for classification. Biodiversity datasets, storing domain knowledge and auxiliary data, can be exploited to develop desired species embedding models (especially when small samples are available for training). These models will then serve as decision support systems for biodiversity researchers to help classify the historical and present-day scientific illustrations from various species of living organisms, which reside underutilised in natural history museums globally.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

We thank Nhung T. H. Nguyen and Sophia Ananiadou (National Centre for Text Mining) for providing the text embeddings from the Biodiversity Heritage Library. This work is supported by the Netherlands Organisation for Scientific Research (NWO) and Brill Publishers, grant 652.001.001.

References

Akata, Z., Reed, S., Walter, D., Lee, H., Schiele, B., 2015a. Evaluation of output embeddings for fine-grained image classification. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, Massachusetts, pp. 2927–2936. https://doi.org/10.1109/CVPR.2015.7298911.

Akata, Z., Perronnin, F., Harchaoui, Z., Schmid, C., 2015b. Label-embedding for image classification. IEEE Trans. Pattern Anal. Mach. Intell. 38 (7), 1425–1438. https://doi.org/10.1109/TPAMI.2015.2487986.

Austen, G.E., Bindemann, M., Griffiths, R.A., Roberts, D.L., 2016. Species identification by experts and non-experts: comparing images from field guides. Sci. Rep. 6, 33634.

Barz, B., Denzler, J., 2019. Hierarchy-based image embeddings for semantic image retrieval. In: Proceedings of the IEEE Winter Conference on Applications of Computer Vision (WACV), Waikoloa Village, Hawaï, pp. 638–647. https://doi.org/10.1109/WACV.2019.00073.

Beery, S., Wu, G., Rathod, V., Votel, R., Huang, J., 2020a. Context r-cnn: long term temporal context for per-camera object detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Seattle, Washington, pp. 13072–13082. https://doi.org/10.1109/CVPR42600.2020.01309.

Beery, S., Cole, E., Gjoka, A., 2020b. The Iwildcam 2020 Competition Dataset.

Belhumeur, P.N., Chen, D., Feiner, S., Jacobs, D.W., Kress, W.J., Ling, H., Lopez, I., Ramamoorthi, R., Sheorey, S., White, S., Zhang, L., 2008. Searching the world's herbaria: A system for visual identification of plant species. In: Proceedings of the European Conference on Computer Vision, Springer, pp. 116–129. https://doi.org/10.1007/978-3-540-88693-8_9.

Berg, T., Liu, J., Woo Lee, S., Alexander, M.L., Jacobs, D.W., Belhumeur, P.N., 2014. Birdsnap: large-scale fine-grained visual categorization of birds. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Columbus, Ohio, pp. 2011–2018. https://doi.org/10.1109/CVPR.2014.259.

Choate, P.M., 1999. Introduction to the Identification of Beetles (Coleoptera), Dichotomous Keys to Some Families of Florida Coleoptera, pp. 23–33.

Chu, G., Potetz, B., Wang, W., Howard, A., Song, Y., Brucher, F., Leung, T., Adam, H., 2019. Geo-aware networks for fine-grained recognition. In: Proceedings of the IEEE International Conference on Computer Vision Workshops, Seoul, Korea (South), pp. 247–254. https://doi.org/10.1109/ICCVW.2019.00033.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., Fei-Fei, L., 2009. Imagenet: a large-scale hierarchical image database. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Miami, Florida, pp. 248–255. https://doi.org/10.1109/CVPR.2009.5206848.

Dietterich, T.G., 1998. Approximate statistical tests for comparing supervised classification learning algorithms. Neural Comput. 10 (7), 1895–1923. https://doi.org/10.1162/089976698300017197.

Drew, J.A., Moreau, C.S., Stiassny, M.L., 2017. Digitization of museum collections holds the potential to enhance researcher diversity. Nature ecology & evolution 1 (12), 1789.
Ferrari, V., Zisserman, A., 2007. Learning visual attributes. In: Advances in Neural Information Processing Systems, pp. 433–440.
Frome, A., Corrado, G.S., Shlens, J., Bengio, S., Dean, J., Ranzato, M., Mikolov, T., 2013. Devise: a deep visual-semantic embedding model. In: Advances in Neural Information Processing Systems, pp. 2121–2129.
GBIF Secretariat, 2018. Gbif Backbone Taxonomy. https://hosted-datasets.gbif.org/datasets/backbone/2018-06-20/.
Gwinn, N.E., Rinaldo, C., 2009. The biodiversity heritage library: sharing biodiversity literature with the world. IFLA J. 35 (1), 25–34. https://doi.org/10.1177/0340035208102032.
Harris, Z.S., 1954. Distributional structure. Word 10 (2–3), 146–162. https://doi.org/10.1080/00437956.1954.11659520.
Hedrick, B.P., Heberling, J.M., Meineke, E.K., Turner, K.G., Grassa, C.J., Park, D.S., Kennedy, J., Clarke, J.A., Cook, J.A., Blackburn, D.C., Edwards, S.V., Davis, C.C., 2020. Digitization and the future of natural history collections. BioScience 70 (3), 243–251. https://doi.org/10.1093/biosci/biz163.
Kumar, N., Belhumeur, P.N., Biswas, A., Jacobs, D.W., Kress, W.J., Lopez, I.C., Soares, J.V.B., 2012. Leafsnap: A computer vision system for automatic plant species identification. In: Fitzgibbon, A., Lazebnik, S., Perona, P., Sato, Y., Schmid, C. (Eds.), Proceedings of the European Conference on Computer Vision. Springer, Berlin, Heidelberg, pp. 502–516. https://doi.org/10.1007/978-3-642-33709-3_36.
Lampert, C.H., Nickisch, H., Harmeling, S., 2009. Learning to detect unseen object classes by between-class attribute transfer. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Miami, Florida, pp. 951–958. https://doi.org/10.1109/CVPR.2009.5206594.
Lampert, C.H., Nickisch, H., Harmeling, S., 2014. Attribute-based classification for zero-shot visual object categorization. IEEE Trans. Pattern Anal. Mach. Intell. 36 (3), 453–465. https://doi.org/10.1109/TPAMI.2013.140.
Maaten, L.V.D., Hinton, G., 2008. Visualizing data using t-sne. Journal of Machine Learning Research 9 (Nov), 2579–2605.
Mac Aodha, O., Cole, E., Perona, P., 2019. Presence-only geographical priors for fine-grained image classification. In: Proceedings of the IEEE International Conference on Computer Vision, Seoul, Korea (South), pp. 9595–9605. https://doi.org/10.1109/ICCV.2019.00969.
Mikolov, T., Sutskever, I., Chen, K., Corrado, G., Dean, J., 2013a. Distributed representations of words and phrases and their compositionality, in: advances in neural information processing systems. Vol. 2, 3111–3119.
Mikolov, T., Chen, K., Corrado, G., Dean, J., 2013b. Efficient estimation of word representations in vector space. In: Workshop proceedings of the International Conference on Learning Representations.
Nguyen, N.T.H., Soto, A.J., Kontonatsios, G., Batista-Navarro, R., Ananiadou, S., 2017. Constructing a biodiversity terminological inventory. PLoS One 12 (4), e0175277. https://doi.org/10.1371/journal.pone.0175277.
Nilsback, M.E., Zisserman, A., 2006. A visual vocabulary for flower classification. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2, pp. 1447–1454. https://doi.org/10.1109/CVPR.2006.42. New York, New York.
Oquab, M., Bottou, L., Laptev, I., Sivic, J., 2014. Learning and transferring mid-level image representations using convolutional neural networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Columbus, Ohio, pp. 1717–1724. https://doi.org/10.1109/CVPR.2014.222.
Patterson, G., Hays, J., 2012. Sun attribute database: discovering, annotating, and recognizing scene attributes. In: Proceedings of the 2012 IEEE Conference on Computer Vision and Pattern Recognition, Providence, Rhode Island, pp. 2751–2758. https://doi.org/10.1109/CVPR.2012.6247998.
Pennington, J., Socher, R., Manning, C., 2014. Glove: Global vectors for word representation. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, pp. 1532–1543. https://doi.org/10.3115/v1/D14-1162.
Romera-Paredes, B., Torr, P.H.S., 2017. An embarrassingly simple approach to zero-shot learning. In: Visual Attributes, Springer, pp. 11–30. https://doi.org/10.1007/978-3-319-50077-5_2.
Snell, J., Swersky, K., Zemel, R., 2017. Prototypical networks for few-shot learning. In: Advances in Neural Information Processing Systems, pp. 4077–4087.
Socher, R., Ganjoo, M., Manning, C.D., Ng, A., 2013. Zero-shot learning through cross-modal transfer. In: Advances in Neural Information Processing Systems, pp. 935–943.
Sumbul, G., Cinbis, R.G., Aksoy, S., 2018. Fine-grained object recognition and zero-shot learning in remote sensing imagery. IEEE Trans. Geosci. Remote Sens. 56 (2), 770–779. https://doi.org/10.1109/TGRS.2017.2754648.
Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., Wojna, Z., 2016. Rethinking the inception architecture for computer vision. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Las Vegas, Nevada, pp. 2818–2826. https://doi.org/10.1109/CVPR.2016.308.
Tsochantaridis, I., Joachims, T., Hofmann, T., Altun, Y., 2005. Large margin methods for structured and interdependent output variables. J. Mach. Learn. Res. 6 (Sep), 1453–1484.
Turney, P.D., Pantel, P., 2010. From frequency to meaning: vector space models of semantics. J. Artif. Intell. Res. 37 (1), 141–188.
Van Horn, G., Branson, S., Farrell, R., Haber, S., Barry, J., Ipeirotis, P., Perona, P., Belongie, S., 2015. Building a bird recognition app and large scale dataset with citizen scientists: the fine print in fine-grained dataset collection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, Massachusetts, pp. 595–604. https://doi.org/10.1109/CVPR.2015.7298658.
Van Horn, G., Mac Aodha, O., Song, Y., Cui, Y., Sun, C., Shepard, A., Adam, H., Perona, P., Belongie, S., 2018. The inaturalist species classification and detection dataset. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, Utah, pp. 8769–8778. https://doi.org/10.1109/CVPR.2018.00914.
Wah, C., Branson, S., Welinder, P., Perona, P., Belongie, S., 2011. The Caltech-UCSD Birds-200-2011 Dataset, Tech. Rep. CNS-TR-2011-001. California Institute of Technology.
Wang, M., Deng, W., 2018. Deep visual domain adaptation: a survey. Neurocomputing 312, 135–153. https://doi.org/10.1016/j.neucom.2018.05.083.
Weber, A., 2020. Collecting colonial nature: European naturalists and the netherlands indies in the early nineteenth century. BMGN-Low Countries Historical Review 134 (3). https://doi.org/10.18352/bmgn-lchr.10741.
Xian, Y., Akata, Z., Sharma, G., Nguyen, Q., Hein, M., Schiele, B., 2016. Latent embeddings for zero-shot classification. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, Nevada, pp. 69–77. https://doi.org/10.1109/CVPR.2016.15.
Xian, Y., Lampert, C.H., Schiele, B., Akata, Z., 2019. Zero-shot learning–a comprehensive evaluation of the good, the bad and the ugly. IEEE Trans. Pattern Anal. Mach. Intell. 41 (9), 2251–2265. https://doi.org/10.1109/TPAMI.2018.2857768.