Webly Supervised Semantic Embeddings for Large Scale Zero-Shot Learning

Yannick Le Cacheux, Adrian Popescu, and Herv Le Borgne

Universit Paris-Saclay, CEA, List, F-91120, Palaiseau, France

Abstract. Zero-shot learning (ZSL) makes object recognition in images possible in absence of visual training data for a part of the classes from a dataset. When the number of classes is large, classes are usually represented by semantic class prototypes learned automatically from unannotated text collections. This typically leads to much lower performances than with manually designed semantic prototypes such as attributes. While most ZSL works focus on the visual aspect and reuse standard semantic prototypes learned from generic text collections, we focus on the problem of semantic class prototype design for large scale ZSL. More specifically, we investigate the use of noisy textual metadata associated to photos as text collections, as we hypothesize they are likely to provide more plausible semantic embeddings for visual classes if exploited appropriately. We thus make use of a source-based filtering strategy to improve the robustness of semantic prototypes. Evaluation on the large scale ImageNet dataset shows a significant improvement in ZSL performances over two strong baselines, and over usual semantic embeddings used in previous works. We show that this improvement is obtained for several embedding methods, leading to state of the art results when one uses automatically created visual and text features.

1 Introduction

Zero-shot learning (ZSL) is useful when an artificial agent needs to recognize classes which have no associated visual data but can be represented by semantic knowledge [1]. The agent is first trained with a set of seen classes, which have visual samples. Then, it needs to recognize instances from either only unseen classes (classical zero-shot learning scenario) or both seen and unseen classes (generalized zero-shot learning). To do so, it has access to visual features and to semantic class prototypes. Most (generalized) zero-shot learning works focus on the proposal of adapted loss functions [2–7] or on the induction of visual features for unseen classes via generative approaches [8–11]. Here, we use standard components for the visual part of the ZSL pipeline and instead study the influence of semantic class prototypes. Early works exploit manually created attributes [12–14] to define these prototypes. While efficient, it requires a very costly annotation effort and is difficult to scale to large datasets. Different strategies were proposed to automate the creation of prototypes in order to tackle large scale ZSL. An early attempt [15] exploited WordNet to extract part attributes. This

method nevertheless assumes that tested datasets can be mapped to WordNet, which is often impossible. The current trend, which leverages advances in natural language processing [16–18], is to exploit standard word embeddings as semantic prototypes. These embeddings are extracted from generic large scale text collections such as Wikipedia [17, 16] or Common Crawl [19, 20]. The advantage of such methods is that prototype creation is based solely on webly supervised or unsupervised collections. However, following [21, 22], only standard embeddings extracted from generic collections were tested in ZSL.

We tackle the creation of semantic class prototypes for large scale ZSL via a method enabling to suitably leverage more adapted text collections for word embedding creation. The standard generic texts are replaced by metadata associated with photo corpora because the latter are more likely to capture relevant visual relations between words. Our method includes processing of the textual content to improve the semantic plausibility of prototypes [20] and exploits a source-based voting strategy to improve robustness of word co-occurrences [23, 24]. We evaluate the proposed approach for automatic building of semantic prototypes using different text collections. We also perform an ablation study to test the robustness with respect to collection size and provide a detailed error analysis. Results for a large scale collection show our approach enables consistent performance improvement compared to existing automatic prototypes. Interesting performance is also obtained for smaller datasets, where the proposed prototypes reduce the gap with manual prototypes. Our contributions can be summarized as follows:

- We focus on the understudied problem of semantic prototype design for ZSL, and propose a method to create better embeddings from noisy tags datasets.
- We conduct extensive experiments and ablation studies to (1) demonstrate the effectiveness of the proposed method; (2) provide a variety of results with different embeddings which can be used for future fair comparison; (3) provide insight on the remaining challenges to close the gap between manual and unsupervised semantic prototypes.
- We collect new corpora and produce state-of-the-art semantic class prototypes for large-scale ZSL which will be released to the community. The code is released at https://github.com/yannick-lc/semantic-embeddings-zsl

2 Related Work

Zero-shot learning. Zero-shot learning [25–28] attempts to classify samples belonging to *unseen classes*, for which no training samples are available. Visual samples are available during training for *seen classes* and both seen classes and unseen classes have "semantic" prototypes associated to them.

The first ZSL approaches were introduced a decade ago [27, 26, 28] and a strong research effort has been devoted to the topic ever since [1, 29, 3, 30–33]. Several of these works relied on a triplet loss to group relevant visual sample close to the prototype in the joint space while discarding irrelevant ones [2, 4–6, 34, 7]. In the generalized zero-shot learning (GZSL) setting, performance is evaluated

both on seen and unseen classes [35]. Then, a strong bias towards recognizing seen classes appears [36]. It is nevertheless possible to tune the hyper-parameters of a ZSL method to boost its performance in a GZSL setting [37]. Recent generative approaches propose to learn discriminative models on unseen classes from artificial samples resulting from a generative model previously learned on seen classes [8–11]. The transductive ZSL setting assumes that the unlabelled visual testing samples can be used during training [38–41]. This usually boosts the performance, but we consider such a hypothesis too restrictive in practice, and this setting is out of the scope here.

Semantic representation. Semantic prototypes can be created either manually or automatically. Since the former are difficult to scale, we focus on automatically created ones, that usually rely on large-scale datasets collected on the Web. The extraction of word representations from the contexts in which they appear is a longstanding topic in natural language processing (NLP). Explicit Semantic Analysis (ESA) [42] is an early attempt to exploit topically structured collections to derive vectorial representations of words. It proposes to represent each word by its tf-idf weights with regard to a large collection of Wikipedia entries (articles). ESA was later improved by adding a temporal aspect to it [43] or by the detection and use of concepts instead of unigrams [44]. ESA and its derivates have good performance in word relatedness and text classification tasks. However, they are relatively difficult to scale because they live in the vectorial space defined by Wikipedia concepts which typically includes millions of entries.

The most influential word representation models in the past years are based on the exploitation of the local context. Compared to ESA, they have the advantage of being orders of magnitude more compact, with typical sizes in the range of hundreds of dimensions. word2vec embeddings [45] are learned from co-occurrences in local context window which are modeled using continuous bag-of-words and skip grams. This model usually outperforms bag-of-words [45, 19, 20]. Some preprocessing steps such as removal of duplicate sentences, phrase detection to replace unigrams, use of subword information or frequent word subsampling is beneficial to the performances [20]. One shortcoming of word embeddings as proposed in [45] is that they only take into account the local context of words. GloVe [18] was introduced as an alternative method which also includes a global component obtained via matrix factorization. The model trains efficiently only on non-zero word-word co-occurrence matrix instead of a sparse matrix or on local windows. It provides superior performance compared to continuous bag-of-words and skip gram models on a series of NLP tasks, including word analogy and similarity. The FastText model [19] derives from that proposed by Mikolov but considers a set of n-grams that can compose the words, compute some embeddings then represent a word as the sum of the vector representation of its n-grams. It thus models the internal structure of the words and allows to compute out of vocabulary word representations. The state of the art in a large array of natural language processing task was recently improved by the introduction of contextual models such as ELMo [46], GPT [47] or BERT [48].

These approaches make use of deep networks and model language at sentence level instead of word level as was the case for skip grams and GloVe. While very interesting for tasks in which words are contextualized, they are not directly applicable to our ZSL scenario which requires the representation of individual words/class names.

Multimodal representations. The word representation approaches presented above exploit only textual resources and there are also attempts to create multimodal word embeddings. Early works projected the vocabulary on a bag-of-visual-words space for image retrieval [49]. More recently, vis-w2v [50] exploits synthetic scenes to learn visual relations between classes. The main challenge here is to model the diversity of natural scenes via synthetic scenes. ViCo [51] exploit word co-occurrences in natural images in order to improve purely textual GloVe embeddings. Visual and textual components complement each other and thus improve performance in tasks such as visual question answering, image retrieval or image captioning. However, an inherent drawback of all these multimodal representations requires representative images of any word to consider and is thus not usable in ZSL for unseen classes. Regarding visual features only, [52, 53] showed that one can train convolutional networks on a dataset of unannotated images collected on the Web, and that these networks perform well in a transfer learning context. Previous works in ZSL used embeddings to represent the semantic prototype, either at a small scale on CUB [54] or at a larger scale on ImageNet, using word2vec [2, 35, 55] (possibly trained on wikipedia [6, 21]), GloVe [34, 22], FastText or ELMo [56]. However, they only use publicly available pre-trained models, while we propose a method to design prototypes that perform better in a ZSL context.

3 Semantic Class Prototypes for Large Scale ZSL

Problem formulation. The zero-shot learning (ZSL) task considers a set $\mathscr{C}_s$ of *seen* classes used during training and a set $\mathscr{C}_u$ of *unseen* classes that are available for the test only. In generalized zero-shot learning (GZSL), additional samples from the seen classes are used for testing as well. However, in both cases, $\mathscr{C}_s \cap \mathscr{C}_u = \emptyset$. Each class has a semantic *class prototype* $\mathbf{s}_c \in \mathbb{R}^K$ that characterizes it. We consider a training set $\{(\mathbf{x}_i, y_i), i = 1 \dots N\}$ with labels $y_i \in \mathscr{C}_s$ and visual features $\mathbf{x}_i \in \mathbb{R}^D$. The task is to learn a compatibility function $f : \mathbb{R}^D \times \mathbb{R}^K \to \mathbb{R}$ assigning a similarity score to a visual sample $\mathbf{x}$ and a class prototype $\mathbf{s}$. f is usually obtained by minimizing a regularized loss function:

$$\frac{1}{N} \sum_{i=1}^{N} \sum_{c=1}^{|\mathscr{C}_s|} \mathscr{L}(f(\mathbf{x}_i, \mathbf{s}_c), y_i) + \lambda \Omega[f] \tag{1}$$

where Ω is a regularization term weighted by λ which constrains the parameters of f, and $\mathscr{L}$ is a loss function. Once a function f is learned, the testing phase

consists in determining the label $\hat{y} \in \mathscr{C}_u$ (or $\hat{y} \in \mathscr{C}_s \cup \mathscr{C}_u$ for GZSL) corresponding to a visual sample $\mathbf{x}$ such that $\hat{y} = \underset{c \in \mathscr{C}_u}{\arg\max}\ f(\mathbf{x}, \mathbf{s}_c)$.

We propose to automatically derive semantic class prototypes $\mathbf{s}_c$ with a method able to adequately leverage noisy corpora which are adapted for visual tasks instead of standard text corpora previously used in ZSL [16, 18, 19]. More specifically, a corpus must contain enough visual information to enable to learn discriminative embeddings. We therefore create two corpora, $\mathbf{fl_{wiki}}$ and $\mathbf{fl_{cust}}$, with this goal in mind.

Corpus collection. $\mathbf{fl_{wiki}}$ is constituted based on Wikipedia. We select salient concepts by ranking English Wikipedia entries by their number of incoming links and keeping the top 120,000 of the list. The default Flickr ranking algorithm is then used to collect up to 5000 photo metadata for each concept. Metadata fields which are exploited here include: (1) *title* – a free text description of the photo (2) *tags* – a list of tags attributed to the photo and (3) the unique identifier of the user. Note that there is no guarantee as to the relevance of textual metadata for the content of each photo since the users are free to upload any text they wish. Also, photo annotations can be made in any language. We illustrate title and tags from Flickr with the following examples: *"smfur Pagophila eburnea Ivory Gull"* and *"minnesota flying inflight gull arctic juvenile duluth rare lakesuperior canalpark ivorygull saintlouiscounty"*. The title includes the Icelandic, Latin and English variants of the name while the tags give indications about the location and activity of the ivory gull. Importantly, tags can be single words (*"gull"*) or concatenated ones (*"ivorygull"*, *"lakesuperior"*). This first collection is made of 62.7 million image metadata pieces and 1.11 billion words.

The fl_{wiki} collection allows to learn generic embeddings that can be used to address large scale ZSL. However, these embeddings are still quite "generic" since they are representative of the Wikipedia concepts. For a given ZSL problem, the visual samples of unseen classes are unknown during training, but the name of these classes can be known before the actual production (testing) phase. Such a hypothesis is implicitly made by most generative ZSL approaches, which synthesize faked visual samples from the prototype only [8–11]. Following a similar hypothesis, we build $\mathbf{fl_{cust}}$, a custom subset of FlickR, which is built using the class names from the three ZSL used in evaluation datasets (ImageNet-ZSL, CUB and AWA). The collection process is similar to that deployed for fl_{wiki}. The only difference is that we use specific class names, which may each have several variants. This collection includes 61.9 million metadata pieces and 995 million unique words.

Each collection therefore consists in a list of $C \leq 120,000$ concepts. For each class c, we have a metadata set $\mathscr{M}_c = \{m_1, \ldots, m_{N_c}\}$ made of $N_c \leq 5,000$ metadata pieces. Each metadata piece m_n consists in a user ID id_n and a list of T_n words $\mathscr{W}_n = \{w_1, \ldots, w_{T_n}\}$, where the words are extracted from titles and tags. T_n is typically in the range of one to two dozens. Note that stop words were discarded during preprocessing.

Creation of embeddings. To create text representations, a vocabulary $\mathscr{V} = \{v_1, \ldots, v_V\}$ is constituted to include all V distinct words in the corpus. We similarly create a set $\mathscr{U} = \{u_1, \ldots, u_U\}$ of all distinct users IDs. The usual skip-gram task [16] aims to find word representations which contain predictive information regarding the words surrounding a given word. Given a sequence $\{w_1, \ldots, w_T\}$ of T training words such that $w_t \in \mathscr{V}$ and a context of size S, the objective is to maximize

$$\sum_{t=1}^{T} \sum_{\substack{-S \leq i \leq S \\ i \neq 0}} \log p(w_{t+i}|w_t) \tag{2}$$

Writing $v_{w_t} \in \mathscr{V}$ the unique word associated with the t^{th} training word w_t and $\mathbf{v}_{w_t}$ and $\mathbf{v}'_{w_t}$ the corresponding "input" and "output" vector representations, $p(w_i|w_t)$ can be computed such that

$$p(w_i|w_t) = \frac{\exp(\mathbf{v}'^{\top}_{w_i} \mathbf{v}_{w_t})}{\sum_{j=1}^{V} \exp(\mathbf{v}'^{\top}_{j} \mathbf{v}_{w_t})} \tag{3}$$

Unlike in standard text collections, such as Wikipedia, the order of words in each metadata collection $\mathscr{M}_n$ is arbitrary. Consequently, using a fixed size window to capture the context of a word is not suitable. We tested the use of fixed size windows in preliminary experiments and results were suboptimal.

Instead, we consider that two words v_i and v_j appear in the same context if both of them appear in the same list of words $\mathscr{W}_n$ of metadata result m_n. The skip-gram objective in Equation 2 can therefore be rewritten as

$$\sum_{c=1}^{C} \sum_{n=1}^{N_c} \sum_{\substack{(v_i, v_j) \\ v_i, v_j \in \mathscr{W}_n,\ i \neq j}} \log p(v_i|v_j) \tag{4}$$

This is equivalent to extracting all pairs of words (v_i, v_j) such that v_i, v_j belong to the same $\mathscr{W}_n$ in a training file, and feeding this resulting corpus to a word embedding model. This has the advantage of enabling the use of available implementations such as word2vec [16] to learn the word embeddings.

Addressing repetitive tags. It is noteworthy that many users perform bulk tagging [24] which consists in attributing the same textual description to a whole set of photos. Users also do semi-bulk, i.e. they attribute a part of tags to an entire photo set and then complete these annotations with photo-specific tags. Bulk is known to bias language models obtained from Flickr [24, 23]. To account for this problem, we add an additional processing step for the two collections. The authors of [23] and [24] suggested to replace simple tag co-occurrences by the number of distinct Flickr users who associated the two words and reported interesting gains in image retrieval and automatic geotagging respectively.

In our case, this translates into adding a pair (v_i, v_j) in the training file only once for each user and thus avoiding the effect of bulk tagging. A positive side effect of filtering pairs with unique users is that the size of the training file is reduced and embeddings are learned faster. A comparison of performance obtained with raw co-occurrence and with user filtering is provided in the supplementary material.

The same ideas can easily be applied to other word embedding approaches. In the next section, we provide experimental results with three such approaches: word2vec [16], GloVe [18] and FastText[19].

4 Experiments

4.1 Evaluation protocol

Baseline methods. To the best of our knowledge, our work is the first to explicitly address the problem of semantic class prototype design for large scale ZSL. We compare to the pre-trained embeddings (noted **pt**), as they are usually used in previous ZSL works [2, 22, 55]. word2vec is trained on Google News with 100 billion words, GloVe is trained on Common Crawl with 840 billion words and the same collection with 600 billion words is used for FastText.

We also propose two baseline methods, (**wiki**) and (**clue**), to which ours can be fairly compared. They consist in learning the embeddings from two different text collections. Wikipedia (**wiki**) is classically exploited to create embeddings because it covers a wide array of topics [42]. *wiki* content is made of entries which describe unambiguous concepts with well formed sentences such as *"The ivory gull is found in the Arctic, in the northernmost parts of Europe and North America."*. The encyclopedia provides good baseline models for a wide variety of tasks [16, 20, 18]. Here we exploit a dump from January 2019 which includes 20.84 billion words. It is the same data as that from which were extracted the $120,000$ concepts for our method. While useful to create transferable embeddings, Wikipedia text does nevertheless not specifically describe visual relations between words. The second baseline is based on visually oriented textual content similar to the one used in our method. The ClueWeb12 [57] collection (**clue**) consists of over 700 million Web pages which were collected so as to cover a wide variety of topics and to avoid spam. We extracted visual metadata from the *title* and *alt* HTML attributes associated to *clue* images. The title content is quite similar to that we extracted from FlickR in our method. *clue* content is often made of short texts such as *"ivory gull flying"* which does not encode a lot of context. After sentence deduplication [20], the resulting collection includes 628 million unique metadata pieces and 3.69 billion words.

Evaluation datasets. The generic object recognition in ZSL requires to be evaluated at a large scale and is thus usually conducted on ImageNet [58]. Frome et al. [2] proposed to use the $1,000$ classes of ILSVRC for training and different subsets of the remaining $20,841$ classes to test. However, it has been recently

showed that a structural bias appears in this setting which allows a "trivial model" to outperform most existing ZSL models [22]. For this reason, we adopt the evaluation protocol proposed by Hascoet *et al.* that considers the same training classes as Frome *et al.* but uses 500 classes with a minimal structural bias for testing [22].

To get insight into the gap existing between manual attributes and unsupervised embeddings, we also conduct experiments on two smaller benchmarks on which the ZSL task is usually conducted with manual attributes specific to each dataset: Caltech UCSD Birds 200-2011 (CUB) [13] and Animals with Attributes 2 (AwA2) [21]. CUB is a fine-grained dataset of 11788 pictures representing 200 bird species and AWA2 a coarse-grained dataset of 37322 pictures depicting 50 animal species. The manual attributes of CUB and AwA2 are respectively 312 and 85-dimensional. In our setting, we are only concerned with semantic prototypes which can be obtained automatically; our results therefore cannot be directly compared to the state-of-the-art algorithms which exploit manual attributes. For CUB and AWA2, we adopt the experimental protocol of Xian *et al.* [21] which relies on *proposed splits* that avoid any overlap between the (unseen) test classes and the ImageNet classes used to pretrain visual features on ILSVRC. For ImageNet, we use the same visual features as [22] while for CUB and AwA2 we adopt those of [21].

ZSL methods. Experiments are conducted with different existing ZSL methods: we provide results for DeViSE [2], ESZSL [3] and ConSE [32] as they are the three standard methods used in [22], and therefore the only methods for which comparable results are currently available. Although results for other models – namely GCN-6 [59], GCN-2 and ADGPM [60] – are also reported in [22], these models are based on graph-convolutional networks [61] which make use of additional intermediate nodes in the WordNet hierarchy. Such methods are outside the scope of this study. We additionally provide results for SynC [6] as well as two linear methods, consisting in a linear projection from the visual to the visual space ($\text{Linear}_{V \to S}$), and a linear projection from the semantic to the visual space ($\text{Linear}_{S \to V}$) inspired by [30], who proposed to compute similarities in the visual space to avoid the hubness problem [62].

We train the models with the usual protocol for ZSL: hyperparameters are determined using a subset of training classes as validation. We sample respectively 200 and 50 such classes at random among the 1000 and 150 training classes of ImageNet and CUB, and use the 8 classes not in ILSVRC among the 40 training classes of AwA2. Since ConSE and DeViSE results depend on a random initialization of the models' parameters, we report results averaged over 5 runs for these two models.

Implementation details. Word embeddings are computed using the original implementations of word2vec [16], GloVe [18] and FastText[19], with the same hyperparameters (see supplementary materials). In particular, we follow the usual text processing steps they propose. Semantic prototypes for all classes

Table 1. ZSL accuracy at large scale (ImageNet dataset), for three embedding models. Each time, the three baselines (*pt*, *wiki* and *clue*) are compared to our method $\mathbf{fl_{wiki}}$ and its variation $\mathbf{fl_{cust}}$. Results marked with "*" correspond to a setting close to Table 2 from Hascoet *et al.* [22], and are consistent with reported results.

Model	word2vec					GloVe					FastText				
Source	*pt*	wiki	clue	$\mathbf{fl_{wiki}}$	$\mathbf{fl_{cust}}$	*pt*	wiki	clue	$\mathbf{fl_{wiki}}$	$\mathbf{fl_{cust}}$	*pt*	wiki	clue	$\mathbf{fl_{wiki}}$	$\mathbf{fl_{cust}}$
$\text{Linear}_{V\to S}$	*6.8*	9.8	9.6	10.5	12.6	*10.2*	6.2	4.2	9.6	9.2	*6.0*	8.9	2.8	11.6	**14.2**
$\text{Linear}_{S\to V}$	*11.6*	11.8	12.2	12.8	17.1	*14.1*	7.9	8.0	9.2	11.4	*14.4*	12.1	8.0	13.3	**17.2**
ESZSL	*10.5*	10.0	10.7	9.5	15.3	*14.1**	8.0	10.3	11.1	12.0	*14.2*	10.1	1.1	11.9	**15.8**
ConSE	*9.9*	10.5	11.3	11.9	13.5	*11.3**	8.1	7.8	11.3	11.9	*11.0*	10.5	5.4	12.6	**14.5**
Devise	*9.0*	9.8	9.9	9.6	13.3	*11.0**	5.9	5.4	3.8	3.4	*12.3*	10.1	5.6	10.3	**13.8**
SynC_{o-vs-o}	*12.2*	12.4	12.6	12.5	16.3	*15.0*	10.9	11.2	12.4	13.3	*14.6*	12.6	7.0	13.2	**16.5**

Table 2. ZSL accuracy at smaller scale with unsupervised semantic class prototypes. Results are reported on the CUB and Awa2 datasets, for three embedding models.

Model	word2vec					GloVe					FastText				
Source	*pt*	wiki	clue	$\mathbf{fl_{wiki}}$	$\mathbf{fl_{cust}}$	*pt*	wiki	clue	$\mathbf{fl_{wiki}}$	$\mathbf{fl_{cust}}$	*pt*	wiki	clue	$\mathbf{fl_{wiki}}$	$\mathbf{fl_{cust}}$
CUB dataset															
$\text{Linear}_{V\to S}$	*7.5*	14.0	13.9	12.2	16.3	*8.0*	11.6	9.8	12.7	14.2	*7.2*	13.8	12.2	11.6	**17.5**
$\text{Linear}_{S\to V}$	*11.3*	18.0	17.2	21.5	23.0	*18.2*	16.0	13.4	14.6	19.0	*16.1*	16.2	16.0	19.9	**24.4**
ESZSL	*15.8*	20.4	17.9	23.0	25.2	*19.9*	17.5	16.9	19.0	20.8	*21.1*	18.7	1.7	23.5	**26.5**
ConSE	*8.3*	19.5	21.6	18.0	21.1	*14.1*	15.1	14.9	16.8	18.4	*14.0*	17.7	19.9	17.6	**23.4**
Devise	*12.6*	17.0	15.8	19.0	19.2	*14.6*	16.3	9.9	18.4	14.8	*16.0*	13.2	13.7	17.4	**22.5**
SynC_{o-vs-o}	*15.3*	19.8	17.3	20.3	21.3	*17.6*	17.2	17.6	21.6	20.5	*17.0*	15.0	15.7	20.2	**24.0**
Awa2 dataset															
$\text{Linear}_{V\to S}$	*31.1*	40.2	38.5	**43.6**	37.9	*40.4*	26.9	34.6	40.5	43.3	*42.1*	39.9	28.1	38.5	41.6
$\text{Linear}_{S\to V}$	*38.1*	44.1	49.7	53.9	55.0	*56.6*	42.4	48.1	41.2	**57.7**	*54.7*	49.3	14.4	50.4	46.5
ESZSL	*40.9*	42.2	55.8	53.1	57.1	***61.4***	37.7	49.0	48.2	44.3	*48.2*	37.6	7.9	49.7	54.6
ConSE	*27.4*	31.3	34.3	**43.3**	39.2	*31.3*	27.4	29.8	38.4	41.4	*34.7*	31.3	16.7	42.3	42.1
Devise	*37.2*	34.1	46.6	33.7	43.4	*43.2*	42.6	44.9	30.6	36.4	***52.0***	40.7	13.5	32.7	37.6
SynC_{o-vs-o}	*43.9*	41.1	45.8	47.1	47.5	*46.9*	46.6	47.4	50.0	52.1	***53.3***	40.0	15.2	45.5	48.1

are computed using the same protocol as [22] for fair comparison. For the same reason, we use the implementation from [22] to run DeViSE, ESZSL, ConSE. We use the implementation from [6] for SynC, and use a custom straightforward implementation for $\text{Linear}_{V\to S}$ and $\text{Linear}_{S\to V}$. All semantic prototypes are $\ell 2$-normalized except with ESZSL to have a setting similar to [22] when applicable. We report results without such a normalization in the supplementary materials, even though the trend is mostly the same.

4.2 Comparison to other approaches

The main results of the evaluation are reported in Table 1 for ImageNet. They confirm the relevance of our method and text collections to learn semantic prototypes for ZSL, as the best results are consistently obtained with our prototypes.

Specifically, for ImageNet, the best result reported on the unbiased split in [22] is 14.1 with ADGPM, and 13.5 with a "traditional" ZSL model (not making use of additional nodes in the class hierarchy), which used GloVe embeddings pretrained on Common Crawl. By contrast, our best result is 17.2 with FastText, obtained with embeddings trained on a much smaller dataset.

We also provide results for CUB and AwA2 in Table 2. These results are less relevant since manual attributes exist for these smaller scale datasets, but still provide interesting insights. Importantly, these results are obtained using *unsupervised* prototypes and should not be directly compared to results obtained with manual attributes. On CUB, the best results are obtained with the embeddings learned on the fl_{cust} collection for the three configurations and significantly outperform previous embeddings. Interestingly, there does not seem to be a clear tendancy on AwA2. It turns out that performance obtainable with unsupervised prototypes on AwA2 is already quite close to performance with manual attributes – see Sec. 4.4. Our method is therefore unable to provide a significant improvement, unlike on the other two datasets.

Within each embeddings methods for all three datasets, the best results are usually obtained with fl_{cust} and fl_{wiki} usually performs better than baseline methods. The gain is especially large when compared to the largest available pretrained models for word2vec and FastText. This result is obtained although the largest collections used to create pretrained embeddings are 2 to 3 orders of magnitude larger than the collections we use. For GloVe on ImageNet, the model pretrained on Common Crawl has the best performance. This embedding has poor behavior for all smaller scale datasets, indicating that the combination of local and global contexts at its core is able to capture interesting information at large scale. While its performance on the smaller pretrained dataset is significantly lower than that of FastText, the two models are nearly equivalent when trained on Common Crawl. A similar finding was reported for text classification tasks [20]. The strong performance of fl_{cust} follows intuition since the collection was specifically built to cover the concepts which appear in the three test dataset. This finding confirms the usefulness of smaller but adapted collections for NLP applications such as medical entity recognition [63] or sentiment analysis [64]. Note that we also combined fl_{wiki} and fl_{cust} to obtain a more generic Flickr model. The obtained results were only marginally better compared to the single use of fl_{cust} and are reported in the supplementary material.

Overall, the best performance is usually obtained with fl_{cust} and FastText embeddings.

4.3 Influence of text collection size

The quality of semantic embeddings is influenced by the size of the text collections used to learn them. Existing comparisons are usually done among different collections [16, 18, 19]. While interesting, these comparisons do not provide direct information about the robustness of each collection. To test robustness, we ablate 50%, 75% and 90% of fl_{cust} and wiki collections and report results for ImageNet using FastText embeddings in Table 3. As expected, performance is

Table 3. ZSL performance with 0%, 50%, 75% and 90% data removed from wiki and fl_{cust} collections, on the ImageNet dataset. With FastText embeddings.

Collection	Data removed	0%	50%	75%	90%
wiki	Linear$_{S\to V}$	12.1	11.6	11.3	10.2
	ESZSL	10.1	9.8	9.9	9.6
	ConSE	10.5	11.0	10.5	9.9
	Devise	10.1	8.3	8.7	8.0
fl_{cust}	Linear$_{S\to V}$	17.2	16.8	16.3	15.6
	ESZSL	15.8	15.1	15.3	14.3
	ConSE	14.5	14.1	14.1	14.3
	Devise	13.8	13.4	13.2	12.5

correlated to the collection size, with the best results being obtained for full text collections and the worst when 90% of them is removed. Interestingly, the performance drop is not drastic for either of the collection. For instance, with only 10% of the initial collections, accuracy drops from 12.1 to 10.2 for *wiki* (15.7% relative drop) and from 17.2 to 15.6 for fl_{cust} (9.3% relative drop). Indeed, according to the Zipf's law, the sorted frequency of words in a language is a decreasing power law. Hence, small corpus contain most of frequent words and increasing their size is useful only to address rare cases. The relative drop is smaller for fl_{cust} compared to wiki, showing that a collection which is adapted for the task is more robust to changes in the quantity of available data.

4.4 Comparison to manual attributes

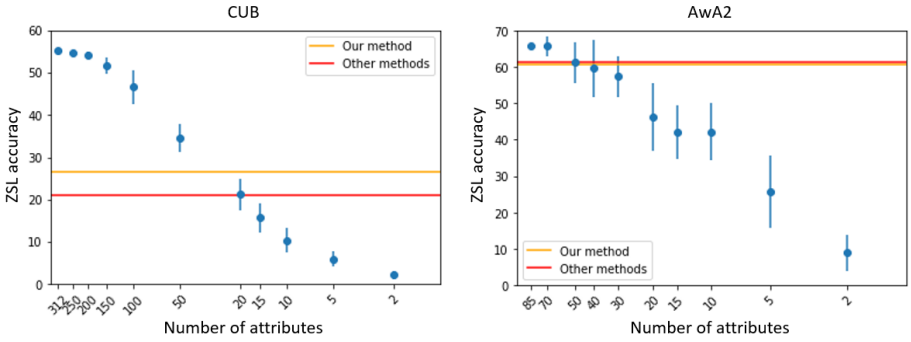

Fig. 1. Ablation of manual attributes on CUB and AwA2. Measured with the linear model, averaged over 10 runs (points and bars are mean±std) with different attributes removed each time. Best results for prototypes based on word embeddings are also reported (horizontal lines).

Although our webly semantic prototypes enable to achieve much better results than with previously available prototypes extracted from text corpora, it is still interesting to compare them to what can be achieved with hand-crafted attributes. Such attributes do not exist for very large scale datasets such as ImageNet, but they are provided with smaller scale datasets such as CUB and AwA2.

To quantify how much better hand-crafted prototypes perform when compared to webly supervised prototypes, we conducted an ablation study on CUB attributes similar to Sec. 4.3. We started with the full list of attributes, initially comprising 312 attributes for each bird species, and randomly removed attributes while measuring the resulting ZSL score. The scores where obtained with the $\text{Linear}_{S\to V}$ model due to its good results, robustness and simplicity. To account for the noise caused by the randomness of the removed attributes, each reported score is the average of 10 measurements, each with different random attributes removed. The remaining attributes are $\ell 2$-normalized, and the hyper-parameter is re-selected by cross-validation for each run. Fig. 1 provides a visualization of the result; a table with the exact scores is available in the supplementary materials.

On CUB, there is still a substantial margin for improvement; even though our method enables a significant increase over other methods, it is still barely above results achievable by selecting only 20 attributes among the 312 initial attributes. Interestingly, the difference between webly supervised and hand-crafted prototypes is not so pronounced on the AwA2 dataset; the ZSL accuracy between the two settings is even surprisingly close. This may be explained by the fact that AwA2 only contains 10 test classes; class prototypes need not enable a ZSL model to subtly distinguish very similar classes. Consequently, our best result is comparable to the best result enabled by previous methods.

4.5 Error analysis

We analyze how far incorrect predictions are from the correct class by computing the distance between the predicted class and the correct class. We define the distance between two classes as the shortest path between them in the WordNet hierarchy. For a given distance d, we measure the number of predictions that are exactly d nodes away from the correct class – a distance of 0 being a correct prediction. Results for $wiki$ and fl_{cust} are presented in Figure 2(a); the general tendency seems to be that classes farther away from the correct class are less likely to be predicted. Note that no two test classes are a distance of one from each other, since it is not possible for a test class to be a direct parent or child of another test class.

We further analyze the main factors behind classification errors. Experiments below are conducted on ImageNet, with the $\text{Linear}_{S\to V}$ model trained using the FastText fl_{cust} embeddings. Our first hypothesis was that the distance between unseen and seen classes influences classification accuracy: the less an unseen class resembles any seen class, the harder it is to identify. To test this hypothesis, we

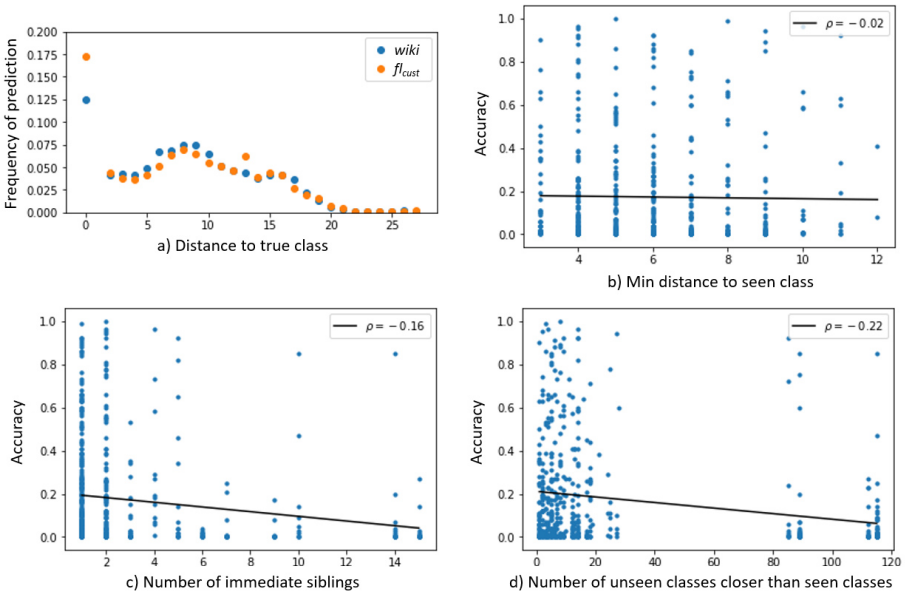

Fig. 2. (a) Distance from predicted class to correct class in the WordNet hierarchy. Correlation ρ between ZSL accuracy and (b) distance to the closest seen class (c) the number of immediate unseen test class siblings (d) the number of unseen classes closer than the closest seen class, for all 500 unseen ImageNet classes.

consider for each unseen class c_u the minimal distance to a seen class $\min_{c \in \mathcal{C}_s} d(c_u, c)$, and analyze its relation to the prediction accuracy. The resulting plot is displayed in Figure 2(b). Surprisingly, the distance to the closest seen class seems to have little to no effect on the accuracy (correlation $\rho = -0.02$).

Another hypothesis was that unseen classes close to other unseen classes are harder to classify than isolated unseen classes, as more confusions are possible. For each unseen class, we therefore compute the number of immediate siblings, a sibling being defined as an unseen class having the same parent in the WordNet hierarchy as the reference (unseen) class. The link between this metric and class accuracy is slightly stronger, with a correlation $\rho = -0.16$ as illustrated in Figure 2(c), but still weak overall.

We combine these two hypotheses by considering the number of unseen classes closer than the closest seen class for each unseen class. The link with class accuracy is more pronounced than by simply considering the number of siblings, with a correlation $\rho = -0.22$ as illustrated in Fig. 2(d). Examples of classes at both ends of the spectrum are visible in Figure 3: unseen class *morel* (on the left) is close to seen class *agaric* and has no unseen siblings; its class accuracy is 0.63. On the other hand, classes *holly*, *teak* and *grevillea* (on the right)

have many unseen siblings and are far from any seen class; their respective accuracy are 0.01, 0.00 and 0.03. More generally, classes which are descendant of the intermediate node *woody plant* have an average accuracy of 0.053. The full graph visualization of the 1000 training classes, 500 testing classes and intermediate nodes of the ImageNet ZSL dataset is provided in the supplementary materials.

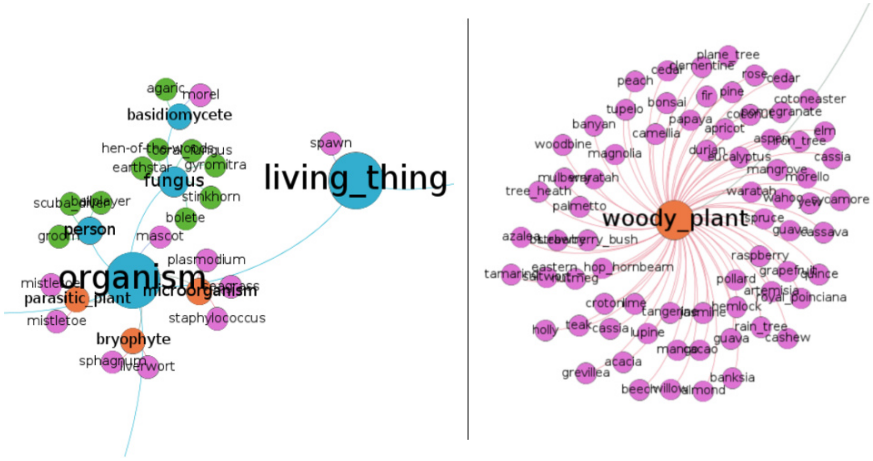

Fig. 3. Graph visualization of parts of the WordNet hierarchy. Green and pink leaves are resp. seen and unseen classes. Intermediate nodes are orange if there is no seen class among their children, and blue otherwise. Full graph is available in the supp. materials.

5 Conclusion

We proposed a new method to build semantic class prototypes automatically, thus enabling to better address large scale ZSL. Our results indicate that appropriately learning embeddings on specialized collections made of photo metadata is better than exploiting generic embeddings as it was done previously in ZSL. This still stands when generic embeddings are learned with collections which are two to three orders of magnitude larger than specialized collections. Among photo metadata based collection, the use of Flickr seems preferable to that of metadata associated to photos from Web pages. This is notably an effect of a better semantic coverage of classes in Flickr compared to ClueWeb12.

References

1. Socher, R., Ganjoo, M., Manning, C.D., Ng, A.: Zero-shot learning through cross-modal transfer. In: Advances in Neural Information Processing Systems. (2013) 935–943
2. Frome, A., Corrado, G.S., Shlens, J., Bengio, S., Dean, J., Mikolov, T., et al.: Devise: A deep visual-semantic embedding model. In: Advances in Neural Information Processing Systems. (2013) 2121–2129
3. Romera-Paredes, B., Torr, P.: An embarrassingly simple approach to zero-shot learning. In: International Conference on Machine Learning. (2015) 2152–2161
4. Akata, Z., Reed, S., Walter, D., Lee, H., Schiele, B.: Evaluation of output embeddings for fine-grained image classification. In: Computer Vision and Pattern Recognition, IEEE (2015) 2927–2936
5. Akata, Z., Perronnin, F., Harchaoui, Z., Schmid, C.: Label-embedding for image classification. Pattern Analysis and Machine Intelligence **38** (2016) 1425–1438
6. Changpinyo, S., Chao, W.L., Gong, B., Sha, F.: Synthesized classifiers for zero-shot learning. In: Computer Vision and Pattern Recognition, IEEE (2016) 5327–5336
7. Annadani, Y., Biswas, S.: Preserving semantic relations for zero-shot learning. In: Computer Vision and Pattern Recognition. (2018) 7603–7612
8. Verma, V.K., Rai, P.: A simple exponential family framework for zero-shot learning. In: Joint European Conference on Machine Learning and Knowledge Discovery in Databases, Springer (2017) 792–808
9. Bucher, M., Herbin, S., Jurie, F.: Zero-shot classification by generating artificial visual features. In: RFIAP. (2018)
10. Verma, V.K., Arora, G., Mishra, A., Rai, P.: Generalized zero-shot learning via synthesized examples. In: Computer Vision and Pattern Recognition. (2018)
11. Xian, Y., Lorenz, T., Schiele, B., Akata, Z.: Feature generating networks for zero-shot learning. In: Computer Vision and Pattern Recognition. (2018)
12. Patterson, G., Hays, J.: Sun attribute database: Discovering, annotating, and recognizing scene attributes. In: Computer Vision and Pattern Recognition, IEEE (2012) 2751–2758
13. Wah, C., Branson, S., Welinder, P., Perona, P., Belongie, S.: The Caltech-UCSD Birds-200-2011 dataset. Technical Report CNS-TR-2011-001, California Institute of Technology (2011)
14. Lampert, C.H., Nickisch, H., Harmeling, S.: Attribute-based classification for zero-shot visual object categorization. Pattern Analysis and Machine Intelligence **36** (2014) 453–465
15. Rohrbach, M., Stark, M., Schiele, B.: Evaluating knowledge transfer and zero-shot learning in a large-scale setting. In: Computer Vision and Pattern Recognition. (2011) 1641–1648
16. Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S., Dean, J.: Distributed representations of words and phrases and their compositionality. In: Advances in Neural Information Processing Systems. (2013) 3111–3119
17. Huang, E., Socher, R., Manning, C., Ng, A.: Improving word representations via global context and multiple word prototypes. In: Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Jeju Island, Korea, Association for Computational Linguistics (2012) 873–882
18. Pennington, J., Socher, R., Manning, C.: Glove: Global vectors for word representation. In: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), Doha, Qatar, Association for Computational Linguistics (2014) 1532–1543

19. Bojanowski, P., Grave, E., Joulin, A., Mikolov, T.: Enriching word vectors with subword information. Transactions of the Association for Computational Linguistics **5** (2017) 135–146
20. Mikolov, T., Grave, E., Bojanowski, P., Puhrsch, C., Joulin, A.: Advances in pre-training distributed word representations. In: Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC-2018), Miyazaki, Japan, European Languages Resources Association (ELRA) (2018)
21. Xian, Y., Lampert, C.H., Schiele, B., Akata, Z.: Zero-shot learning-a comprehensive evaluation of the good, the bad and the ugly. Pattern Analysis and Machine Intelligence (2018)
22. Hascoet, T., Ariki, Y., Takiguchi, T.: On zero-shot recognition of generic objects. In: Computer Vision and Pattern Recognition. (2019)
23. Popescu, A., Grefenstette, G.: Social media driven image retrieval. In: Proceedings of the 1st ACM International Conference on Multimedia Retrieval. ICMR '11, New York, NY, USA, ACM (2011) 33:1–33:8
24. O'Hare, N., Murdock, V.: Modeling locations with social media. Information Retrieval **16** (2013) 30–62
25. Akata, Z., Perronnin, F., Harchaoui, Z., Schmid, C.: Label-embedding for attribute-based classification. In: Computer Vision and Pattern Recognition. (2013) 819–826
26. Lampert, C.H., Nickisch, H., Harmeling, S.: Learning to detect unseen object classes by between-class attribute transfer. In: Computer Vision and Pattern Recognition, IEEE (2009) 951–958
27. Larochelle, H., Erhan, D., Bengio, Y.: Zero-data learning of new tasks. In: AAAI Conference on Artificial Intelligence. (2008)
28. Palatucci, M., Pomerleau, D., Hinton, G.E., Mitchell, T.M.: Zero-shot learning with semantic output codes. In: Advances in Neural Information Processing Systems. (2009) 1410–1418
29. Zhang, Z., Saligrama, V.: Zero-shot learning via semantic similarity embedding. In: International Conference on Computer Vision. (2015) 4166–4174
30. Shigeto, Y., Suzuki, I., Hara, K., Shimbo, M., Matsumoto, Y.: Ridge regression, hubness, and zero-shot learning. In: Joint European Conference on Machine Learning and Knowledge Discovery in Databases, Springer (2015) 135–151
31. Rahman, S., Khan, S., Porikli, F.: A unified approach for conventional zero-shot, generalized zero-shot, and few-shot learning. IEEE Transactions on Image Processing **27** (2018) 5652–5667
32. Norouzi, M., Mikolov, T., Bengio, S., Singer, Y., Shlens, J., Frome, A., Corrado, G., Dean, J.: Zero-shot learning by convex combination of semantic embeddings. In: International Conference on Learning Representations. (2014)
33. Xian, Y., Akata, Z., Sharma, G., Nguyen, Q., Hein, M., Schiele, B.: Latent embeddings for zero-shot classification. In: Computer Vision and Pattern Recognition. (2016) 69–77
34. Changpinyo, S., Chao, W.L., Gong, B., Sha, F.: Classifier and exemplar synthesis for zero-shot learning. arXiv preprint arXiv:1812.06423 (2018)
35. Chao, W.L., Changpinyo, S., Gong, B., Sha, F.: An empirical study and analysis of generalized zero-shot learning for object recognition in the wild. In: European Conference on Computer Vision, Springer (2016) 52–68
36. Xian, Y., Schiele, B., Akata, Z.: Zero-shot learning – the good, the bad and the ugly. In: Computer Vision and Pattern Recognition, IEEE (2017) 3077–3086
37. Le Cacheux, Y., Le Borgne, H., Crucianu, M.: From classical to generalized zero-shot learning: A simple adaptation process. In: International Conference on Multimedia Modeling, Springer (2019) 465–477

38. Fu, Y., Hospedales, T.M., Xiang, T., Gong, S.: Transductive multi-view zero-shot learning. Pattern Analysis and Machine Intelligence **37** (2015) 2332–2345
39. Kodirov, E., Xiang, T., Fu, Z., Gong, S.: Unsupervised domain adaptation for zero-shot learning. In: International Conference on Computer Vision. (2015) 2452–2460
40. Rohrbach, M., Ebert, S., Schiele, B.: Transfer learning in a transductive setting. In: Advances in Neural Information Processing Systems. (2013) 46–54
41. Song, J., Shen, C., Yang, Y., Liu, Y., Song, M.: Transductive unbiased embedding for zero-shot learning. In: Computer Vision and Pattern Recognition. (2018) 1024–1033
42. Gabrilovich, E., Markovitch, S.: Computing semantic relatedness using wikipedia-based explicit semantic analysis. In: Proceedings of the 20th International Joint Conference on Artifical Intelligence. IJCAI'07, San Francisco, CA, USA, Morgan Kaufmann Publishers Inc. (2007) 1606–1611
43. Radinsky, K., Agichtein, E., Gabrilovich, E., Markovitch, S.: A word at a time: Computing word relatedness using temporal semantic analysis. In: Proceedings of the 20th International Conference on World Wide Web. WWW '11, New York, NY, USA, ACM (2011) 337–346
44. Hassan, S., Mihalcea, R.: Semantic relatedness using salient semantic analysis. In: Proceedings of the Twenty-Fifth AAAI Conference on Artificial Intelligence. AAAI'11, AAAI Press (2011) 884–889
45. Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S., Dean, J.: Distributed representations of words and phrases and their compositionality. In Burges, C.J.C., Bottou, L., Welling, M., Ghahramani, Z., Weinberger, K.Q., eds.: Advances in Neural Information Processing Systems 26. Curran Associates, Inc. (2013) 3111–3119
46. Peters, M., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., Zettlemoyer, L.: Deep contextualized word representations. In: Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers), New Orleans, Louisiana, Association for Computational Linguistics (2018) 2227–2237
47. Radford, A., Narasimhan, K., Salimans, T., Sutskever, I.: Improving language understandingby generative pre-training (2018)
48. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: BERT: Pre-training of deep bidirectional transformers for language understanding. In: Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), Minneapolis, Minnesota, Association for Computational Linguistics (2019) 4171–4186
49. Znaidia, A., Shabou, A., Le Borgne, H., Hudelot, C., Paragios, N.: Bag-of-multimedia-words for image classification. In: Pattern Recognition (ICPR), 2012 21st International Conference on, IEEE (2012) 1509–1512
50. Kottur, S., Vedantam, R., Moura, J.M.F., Parikh, D.: Visual word2vec (vis-w2v): Learning visually grounded word embeddings using abstract scenes. In: Computer Vision and Pattern Recognition. (2016)
51. Gupta, T., Schwing, A., Hoiem, D.: Vico: Word embeddings from visual co-occurrences. In: International Conference on Computer Vision. (2019)
52. Joulin, A., van der Maaten, L., Jabri, A., Vasilache, N.: Learning visual features from large weakly supervised data. In: European Conference on Computer Vision, Springer International Publishing (2016) 67–84
53. Vo, P., Ginsca, A.L., Le Borgne, H., Popescu, A.: Harnessing noisy web images for deep representation. Computer Vision and Image Understanding (2017) on line jan 2017.

54. Akata, Z., Malinowski, M., Fritz, M., Schiele, B.: Multi-cue zero-shot learning with strong supervision. In: Computer Vision and Pattern Recognition. (2016)
55. Le Cacheux, Y., Le Borgne, H., Crucianu, M.: Modeling inter and intra-class relations in the triplet loss for zero-shot learning. In: International Conference on Computer Vision, Seoul, Korea (2019)
56. Le Cacheux, Y., Le Borgne, H., Crucianu, M.: Using sentences as semantic representations in large scale zero-shot learning. In: ECCV Task-CV workshop. (2020)
57. Callan, J.: The lemur projectand its clueweb12 dataset (2012)
58. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: Computer Vision and Pattern Recognition, IEEE (2009) 248–255
59. Wang, X., Ye, Y., Gupta, A.: Zero-shot recognition via semantic embeddings and knowledge graphs. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2018) 6857–6866
60. Kampffmeyer, M., Chen, Y., Liang, X., Wang, H., Zhang, Y., Xing, E.P.: Rethinking knowledge graph propagation for zero-shot learning. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2019) 11487–11496
61. Kipf, T.N., Welling, M.: Semi-supervised classification with graph convolutional networks. arXiv preprint arXiv:1609.02907 (2016)
62. Radovanović, M., Nanopoulos, A., Ivanović, M.: Hubs in space: Popular nearest neighbors in high-dimensional data. Journal of Machine Learning Research **11** (2010) 2487–2531
63. El Boukkouri, H., Ferret, O., Lavergne, T., Zweigenbaum, P.: Embedding strategies for specialized domains: Application to clinical entity recognition. In: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop, Florence, Italy, Association for Computational Linguistics (2019) 295–30
64. Kameswara Sarma, P., Liang, Y., Sethares, B.: Domain adapted word embeddings for improved sentiment classification. In: Proceedings of the Workshop on Deep Learning Approaches for Low-Resource NLP, Melbourne, Association for Computational Linguistics (2018) 51–59