Zero-Shot Learning by Convex Combination of Semantic Embeddings

Mohammad Norouzi*, **Tomas Mikolov**, **Samy Bengio**, **Yoram Singer**, **Jonathon Shlens**, **Andrea Frome**, **Greg S. Corrado**, **Jeffrey Dean**

norouzi@cs.toronto.edu, {tmikolov, bengio, singer}@google.com
{shlens, afrome, gcorrado, jeff}@google.com

*University of Toronto, ON, Canada

Google, Inc., Mountain View, CA, USA

Abstract

Several recent publications have proposed methods for mapping images into continuous semantic embedding spaces. In some cases the embedding space is trained jointly with the image transformation. In other cases the semantic embedding space is established by an independent natural language processing task, and then the image transformation into that space is learned in a second stage. Proponents of these image embedding systems have stressed their advantages over the traditional n-way classification framing of image understanding, particularly in terms of the promise for zero-shot learning – the ability to correctly annotate images of previously unseen object categories. In this paper, we propose a simple method for constructing an image embedding system from any existing n-way image classifier and a semantic word embedding model, which contains the n class labels in its vocabulary. Our method maps images into the semantic embedding space via convex combination of the class label embedding vectors, and requires no additional training. We show that this simple and direct method confers many of the advantages associated with more complex image embedding schemes, and indeed outperforms state of the art methods on the ImageNet zero-shot learning task.

1 Introduction

The classic machine learning approach to object recognition presupposes the existence of a large *labeled* training dataset to optimize the free parameters of an image classifier. There have been continued efforts in collecting larger image corpora with a broader coverage of object categories (*e.g.,* [3]), thereby enabling image classification with many classes. While annotating more object categories in images can lead to a finer granularity of image classification, creating high quality fine grained image annotations is challenging, expensive, and time consuming. Moreover, as new visual entities emerge over time, the annotations should be revised, and the classifiers should be re-trained.

Motivated by the challenges facing standard machine learning framework for n-way classification, especially when n (the number of class labels) is large, several recent papers have proposed methods for mapping images into semantic embedding spaces [14, 4, 9, 6, 18, 19]. In doing so, it is hoped that by resorting to nearest neighbor search in the embedding space with respect to a set of label embedding vectors, one can address *zero-shot learning* – annotation of images with new labels corresponding to previously unseen object categories. While the common practice for image embedding is to learn a regression model from images into a semantic embedding space, it has been unclear whether there exists a more direct way to transform any probabilistic n-way classifier into

*Part of this work was done while Mohammad Norouzi was at Google.

an image embedding model, which can be used for zero-shot learning. In this work, we present a simple method for constructing an image embedding system by combining any existing probabilistic n-way image classifier with an existing word embedding model, which contains the n class labels in its vocabulary. We show that our simple method confers many of the advantages associated with more complex image embedding schemes.

Recently, zero-shot learning [10, 14] has received a growing amount of attention [16, 11, 6, 18]. A key to zero-shot learning is the use of a set of semantic embedding vectors associated with the class labels. These semantic embedding vectors might be obtained from human-labeled object attributes [4, 9], or they might be learned from a text corpus in an unsupervised fashion, based on an independent natural language modeling task [6, 18, 12]. Regardless of the way the label embedding vectors are obtained, previous work casts zero-shot learning as a regression problem from the input space into the embedding space. In contrast, given a pre-trained standard classifier, our method maps images into the semantic embedding space via the convex combination of the class label embedding vectors. The values of a given classifier's predictive probabilities for different training labels are used to compute a weighted combination of the label embeddings in the semantic space. This provides a continuous embedding vector for each image, which is then used for extrapolating the pre-trained classifier's predictions beyond the *training* labels, into a set of *test* labels.

The effectiveness of our method called "convex combination of semantic embeddings" (ConSE) is evaluated on ImageNet zero-shot learning task. By employing a convolutional neural network [7] trained only on 1000 object categories from ImageNet, the ConSE model is able to achieve 9.4% hit@1 and 24.7% hit@5 on 1600 unseen objects categories, which were omitted from the training dataset. When the number of test classes gets larger, and they get further from the training classes in the ImageNet category hierarchy, the zero-shot classification results get worse, as expected, but still our model outperforms a recent state-of-the-art model [6] applied to the same task.

2 Previous work

Zero-shot learning is closely related to *one-shot learning* [13, 5, 1, 8], where the goal is to learn object classifiers based on a few labeled training exemplars. The key difference in zero-shot learning is that no training images are provided for a held-out set of test categories. Thus, zero-shot learning is more challenging, and the use of side information about the interactions between the class labels is more essential in this setting. Nevertheless, we expect that advances in zero-shot learning will benefit one-shot learning, and visual recognition in general, by providing better ways to incorporate prior knowledge about the relationships between the object categories.

A key component of zero-shot learning is the way a semantic space of class label embeddings is defined. In computer vision, there has been a body of work on the use of human-labeled visual attributes [4, 9] to help detecting unseen object categories. *Binary* attributes are most commonly used to encode presence and absence of a set of visual characteristics within instances of an object category. Some examples of these attributes include different types of materials, different colors, textures, and object parts. More recently, relative attributes [15] are proposed to strengthen the attribute based representations. In attribute based approaches, each class label is represented by a vector of attributes, instead of the standard one-of-n encoding. And multiple classifiers are trained for predicting each object attribute. While this is closely related to our approach, the main issue with attribute-based classification is its lack of scalability to large-scale tasks. Annotating thousands of attributes for thousands of object classes is an ambiguous and challenging task in itself, and the applicability of supervised attributes to large-scale zero-shot learning is limited. There has been some recent work showing good zero-shot classification performance on visual recognition tasks [17, 11], but these methods also rely on the use of knowledge bases containing descriptive properties of object classes, and the WordNet hierarchy.

A more scalable approach to semantic embeddings of class labels builds upon the recent advances in unsupervised neural language modeling [2]. In this approach, a set of multi-dimensional embedding vectors are learned for each word in a text corpus. The word embeddings are optimized to increase the predictability of each word given its context [12]. Essentially, the words that cooccur in similar contexts, are mapped onto similar embedding vectors. Frome et al. [6] and Socher et al. [18] exploit such word embeddings to embed textual names of object class labels into a continuous semantic space. In this work, we also use the skip-gram model [12] to learn the class label embeddings.

3 Problem Statement

Assume that a labeled training dataset of images $\mathcal{D}_0 \equiv \{(\mathbf{x}_i, y_i)\}_{i=1}^m$ is given, where each image is represented by a p-dimensional feature vector, $\mathbf{x}_i \in \mathbb{R}^p$. For generality we denote $\mathcal{X} \stackrel{\text{def}}{=} \mathbb{R}^p$. There are n_0 distinct class labels available for training, *i.e.,* $y_i \in \mathcal{Y}_0 \equiv \{1, \ldots, n_0\}$. In addition, a test dataset denoted $\mathcal{D}_1 \equiv \{(\mathbf{x}'_j, y'_j)\}_{j=1}^{m'}$ is provided, where $\mathbf{x}'_j \in \mathcal{X}$ as above, while $y'_j \in \mathcal{Y}_1 \equiv \{n_0+1, \ldots, n_0+n_1\}$. The test set contains n_1 distinct class labels, which are omitted from the training set. Let $n = n_0+n_1$ denote the total number of labels in the training and test sets.

The goal of zero-shot learning is to train a classifier on the training set $\mathcal{D}_0$, which performs reasonably well on the unseen test set $\mathcal{D}_1$. Clearly, without any side information about the relationships between the labels in $\mathcal{Y}_0$ and $\mathcal{Y}_1$, zero-shot learning is infeasible as $\mathcal{Y}_0 \cap \mathcal{Y}_1 = \varnothing$. However, to mitigate zero-shot learning, one typically assumes that each class label y ($1 \leq y \leq n$) is associated with a semantic embedding vector $s(y) \in \mathcal{S} \equiv \mathbb{R}^q$. The semantic embedding vectors are such that two labels y and y' are similar if and only if their semantic embeddings $s(y)$ and $s(y')$ are close in $\mathcal{S}$, *e.g.,* $\langle s(y), s(y')\rangle_{\mathcal{S}}$ is large. Clearly, given an embedding of training and test class labels into a joint semantic space *i.e.,* $\{\mathbf{s}(y);\ y \in \mathcal{Y}_0 \cup \mathcal{Y}_1\}$, the training and test labels become related, and one can hope to learn from the training labels to predict the test labels.

Previous work (*e.g.,* [6, 18]) has addressed zero-shot classification by learning a mapping from input features to semantic label embedding vectors using a multivariate regression model. Accordingly, during training instead of learning an n_0-way classifier from inputs to training labels ($\mathcal{X} \rightarrow \mathcal{Y}_0$), a regression model is learned from inputs to the semantic embedding space ($\mathcal{X} \rightarrow \mathcal{S}$). A training dataset of inputs paired with semantic embeddings, *i.e.,* $\{(\mathbf{x}_i, s(y_i));\ (\mathbf{x}_i, y_i) \in \mathcal{D}_0\}$, is constructed to train a regression function $f : \mathcal{X} \rightarrow \mathcal{S}$ that aims to map $\mathbf{x}_i$ to $s(y_i)$. Once $f(\cdot)$ is learned, it is applied to a test image $\mathbf{x}'_j$ to obtain $f(\mathbf{x}'_j)$, and this continuous semantic embedding for $\mathbf{x}'_j$ is then compared with the test label embedding vectors, $\{s(y');\ y' \in \mathcal{Y}_1\}$, to find the most relevant test labels. Thus, instead of directly mapping from $\mathcal{X} \rightarrow \mathcal{Y}_1$, which seems impossible, zero-shot learning methods first learn a mapping $\mathcal{X} \rightarrow \mathcal{S}$, and then a deterministic mapping such as k-nearest neighbor search in the semantic space is used to map a point in $\mathcal{S}$ to a ranked list of labels in $\mathcal{Y}_1$.

4 ConSE: Convex combination of semantic embeddings

4.1 Model Description

In contrast to previous work which casts zero-shot learning as a regression problem from the input space to the semantic label embedding space, in this work, we do not explicitly learn a regression function $f : \mathcal{X} \rightarrow \mathcal{S}$. Instead, we follow the classic machine learning approach, and learn a classifier from training inputs to training labels. To this end, a classifier p_0 is trained on $\mathcal{D}_0$ to estimate the probability of an image $\mathbf{x}$ belonging to a class label $y \in \mathcal{Y}_0$, denoted $p_0(y \mid \mathbf{x})$, where $\sum_{y=1}^{n_0} p_0(y \mid \mathbf{x}) = 1$. Given p_0, we propose a method to transfer the probabilistic predictions of the classifier beyond the training labels, to a set of test labels.

Let $\widehat{y}_0(\mathbf{x}, 1)$ denote the most likely training label for an image $\mathbf{x}$ according to the classifier p_0. Formally, we denote

$$\widehat{y}_0(\mathbf{x}, 1) \equiv \operatorname*{argmax}_{y \in \mathcal{Y}_0} p_0(y \mid \mathbf{x}) \ . \tag{1}$$

Analogously, let $\widehat{y}_0(\mathbf{x}, t)$ denote the t^{th} most likely training label for $\mathbf{x}$ according to p_0. In other words, $p_0(\widehat{y}_0(\mathbf{x}, t) \mid \mathbf{x})$ is the t^{th} largest value among $\{p_0(y \mid \mathbf{x});\ y \in \mathcal{Y}_0\}$. Given the top T predictions of p_0 for an input $\mathbf{x}$, our model deterministically predicts a semantic embedding vector $f(\mathbf{x})$ for an input $\mathbf{x}$, as the convex combination of the semantic embeddings $\{s(\widehat{y}_0(\mathbf{x}, t))\}_{t=1}^T$ weighted by their corresponding probabilities. More formally,

$$f(\mathbf{x}) = \frac{1}{Z} \sum_{t=1}^{T} p(\widehat{y}_0(\mathbf{x}, t) \mid \mathbf{x}) \cdot s(\widehat{y}_0(\mathbf{x}, t)) \ , \tag{2}$$

where Z is a normalization factor given by $Z = \sum_{t=1}^T p(\widehat{y}_0(\mathbf{x}, t) \mid \mathbf{x})$, and T is a hyper-parameter controlling the maximum number of embedding vectors to be considered. If the classifier is very confident in its prediction of a label y for $\mathbf{x}$, *i.e.,* $p_0(y \mid \mathbf{x}) \approx 1$, then $f(\mathbf{x}) \approx s(y)$. However, if the classifier have doubts whether an image contains a "lion" or a "tiger", *e.g.,* $p_0(\text{lion} \mid \mathbf{x}) = 0.6$ and $p_0(\text{tiger} \mid \mathbf{x}) = 0.4$, then our predicted semantic embedding, $f(\mathbf{x}) = 0.6 \cdot s(\text{lion}) + 0.4 \cdot s(\text{tiger})$, will be something between lion and tiger in the semantic space. Even though "liger" (a hybrid cross between a lion and a tiger) might not be among the training labels, because it is likely that $s(\text{liger}) \approx \frac{1}{2}s(\text{lion}) + \frac{1}{2}s(\text{tiger})$, then it is likely that $f(\mathbf{x}) \approx s(\text{liger})$.

Given the predicted embedding of $\mathbf{x}$ in the semantic space, *i.e.,* $f(\mathbf{x})$, we perform zero-shot classification by finding the class labels with embeddings nearest to $f(\mathbf{x})$ in the semantic space. The top prediction of our model for an image $\mathbf{x}$ from the test label set, denoted $\widehat{y}_1(\mathbf{x}, 1)$, is given by

$$\widehat{y}_1(\mathbf{x}, 1) \equiv \underset{y' \in \mathcal{Y}_1}{\text{argmax}}\, cos(f(\mathbf{x}), s(y'))\ , \tag{3}$$

where we use cosine similarity to rank the embedding vectors. Moreover, let $\widehat{y}_1(\mathbf{x}, k)$ denote the k^{th} most likely test label predicted for $\mathbf{x}$. Then, $\widehat{y}_1(\mathbf{x}, k)$ is defined as the label $y' \in \mathcal{Y}_1$ with the k^{th} largest value of cosine similarity in $\{cos(f(\mathbf{x}), s(y'));\ y' \in \mathcal{Y}_1\}$. Note that previous work on zero-shot learning also uses a similar k-nearest neighbor procedure in the semantic space to perform label extrapolation. The key difference in our work is that we define the embedding prediction $f(\mathbf{x})$ based on a standard classifier as in Eq. (2), and not based on a learned regression model. For the specific choice of cosine similarity to measure closeness in the embedding space, the norm of $f(\mathbf{x})$ does not matter, and we could drop the normalization factor $(1/Z)$ in Eq. (2).

4.2 Difference with DeViSE

Our model is inspired by a technique recently proposed for image embedding, called "Deep Visual-Semantic Embedding" (DeViSE) [6]. Both DeViSE and ConSE models benefit from the convolutional neural network classifier of Krizhevsky et al. [7], but there is an important difference in the way they employ the neural net. The DeViSE model replaces the last layer of the convolutional net, the Softmax layer, with a linear transformation layer. The new transformation layer is trained using a ranking objective to map training inputs close to continuous embedding vectors corresponding to correct labels. Subsequently, the lower layers of the convolutional neural network are fine-tuned using the ranking objective to produce better results. In contrast, the ConSE model keeps the Softmax layer of the convolutional net intact, and it does not train the neural network any further. Given a test image, the ConSE simply runs the convolutional classifier and considers the top T predictions of the model. Then, the convex combination of the corresponding T semantic embedding vectors in the semantic space (see Eq. (2)) is computed, which defines a deterministic transformation from the outputs of the Softmax classifier into the embedding space.

5 Experiments

We compare our approach, "convex combination of semantic embedding" (**ConSE**), with a state-of-the-art method called "Deep Visual-Semantic Embedding" (**DeViSE**) [6] on the ImageNet dataset [3]. Both of the ConSE and DeViSE models make use of the same skipgram text model [12] to define the semantic label embedding space. The skipgram model was trained on 5.4 billion words from Wikipedia.org to construct 500 dimensional word embedding vectors. The embedding vectors are then normalized to have a unit norm. The convolutional neural network of [7], used in both ConSE and DeViSE, is trained on ImageNet 2012 1K set with 1000 training labels. Because the image classifier, and the label embedding vectors are identical in the ConSE and DeViSE models, we can perform a direct comparison between the two embedding techniques.

We mirror the ImageNet zero-shot learning experiments of [6]. Accordingly, we report the zero-shot generalization performance of the models on three test datasets with increasing degree of difficulty. The first test dataset, called "2-hops" includes labels from the 2011 21K set which are visually and semantically similar to the training labels in the ImageNet 2012 1K set. This dataset only includes labels within 2 tree hops of the ImageNet 2012 1K labels. A more difficult dataset including labels within 3 hops of the training labels is created in a similar way and referred to as "3-hops". Finally, a dataset of all the labels in the ImageNet 2011 21K set is created. The three test datasets respectively include $1,589$, $7,860$, and $20,900$ labels. These test datasets do not include any image labeled with any of the 1000 training labels.

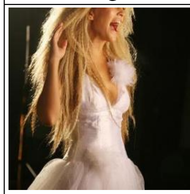
Test Image	Softmax Baseline [7]	DeViSE [6]	ConSE (10)
	wig fur coat Saluki, gazelle hound Afghan hound, Afghan stole	water spaniel tea gown bridal gown, wedding gown spaniel tights, leotards	business suit **dress, frock** hairpiece, false hair, postiche swimsuit, swimwear, bathing suit kit, outfit
	ostrich, Struthio camelus black stork, Ciconia nigra vulture crane peacock	heron owl, bird of Minerva, bird of night hawk bird of prey, raptor, raptorial bird finch	**ratite, ratite bird, flightless bird** peafowl, bird of Juno common spoonbill New World vulture, cathartid Greek partridge, rock partridge
	sea lion plane, carpenter's plane cowboy boot loggerhead, loggerhead turtle goose	elephant turtle turtleneck, turtle, polo-neck flip-flop, thong handcart, pushcart, cart, go-cart	California sea lion **Steller sea lion** Australian sea lion South American sea lion eared seal
	hamster broccoli Pomeranian capuchin, ringtail weasel	**golden hamster, Syrian hamster** rhesus, rhesus monkey pipe shaker American mink, Mustela vison	**golden hamster, Syrian hamster** rodent, gnawer Eurasian hamster rhesus, rhesus monkey rabbit, coney, cony
 **(farm machine)**	thresher, threshing machine tractor harvester, reaper half track snowplow, snowplough	truck, motortruck skidder tank car, tank automatic rifle, machine rifle trailer, house trailer	flatcar, flatbed, flat truck, motortruck tracked vehicle bulldozer, dozer wheeled vehicle
 **(alpaca, Lama pacos)**	Tibetan mastiff titi, titi monkey koala, koala bear, kangaroo bear llama chow, chow chow	kernel littoral, litoral, littoral zone, sands carillon Cabernet, Cabernet Sauvignon poodle, poodle dog	dog, domestic dog domestic cat, house cat schnauzer Belgian sheepdog domestic llama, Lama peruana

Figure 1: Zero-shot test images from ImageNet, and their corresponding top 5 labels predicted by the Softmax Baseline [7], DeViSE [6], and ConSE($T = 10$). The labels predicted by the Softmax baseline are the labels used for training, and the labels predicted by the other two models are not seen during training of the image classifiers. The correct labels are shown in blue. Examples are hand-picked to illustrate the cases that the ConSE(10) performs well, and a few failure cases.

Fig. 1 depicts some qualitative results. The first column shows the top 5 predictions of the convolutional net, referred to as the Softmax baseline [7]. The second and third columns show the zero-shot predictions by the DeViSE and ConSE(10) models. The ConSE(10) model uses the top $T = 10$ predictions of the Softmax baseline to generate convex combination of embeddings. Fig. 1 shows that the labels predicted by the ConSE(10) model are generally coherent and they include very few outliers. In contrast, the top 5 labels predicted by the DeViSE model include more outliers such as "flip-flop" predicted for a "Steller sea lion", "pipe" and "shaker" predicted for a "hamster", and "automatic rifle" predicted for a "farm machine".

Test Label Set	# Candidate Labels	Model	Flat hit@k (%)				
			1	2	5	10	20
2-hops	$1,589$	DeViSE	6.0	10.0	18.1	26.4	36.4
		ConSE(1)	9.3	14.4	23.7	30.8	38.7
		ConSE(10)	**9.4**	**15.1**	**24.7**	**32.7**	**41.8**
		ConSE(1000)	9.2	14.8	24.1	32.1	41.1
2-hops (+1K)	$1,589$ $+1000$	DeViSE	**0.8**	2.7	7.9	14.2	22.7
		ConSE(1)	0.2	**7.1**	**17.2**	24.0	31.8
		ConSE(10)	0.3	6.2	17.0	**24.9**	**33.5**
		ConSE(1000)	0.3	6.2	16.7	24.5	32.9
3-hops	$7,860$	DeViSE	1.7	2.9	5.3	8.2	12.5
		ConSE(1)	2.6	4.2	7.3	10.8	14.8
		ConSE(10)	**2.7**	**4.4**	**7.8**	**11.5**	**16.1**
		ConSE(1000)	2.6	4.3	7.6	11.3	15.7
3-hops (+1K)	$7,860$ $+1000$	DeViSE	**0.5**	1.4	3.4	5.9	9.7
		ConSE(1)	0.2	**2.4**	**5.9**	9.3	13.4
		ConSE(10)	0.2	2.2	**5.9**	**9.7**	**14.3**
		ConSE(1000)	0.2	2.2	5.8	9.5	14.0
ImageNet 2011 21K	$20,841$	DeViSE	0.8	1.4	2.5	3.9	6.0
		ConSE(1)	1.3	2.1	3.6	5.4	7.6
		ConSE(10)	**1.4**	**2.2**	**3.9**	**5.8**	**8.3**
		ConSE(1000)	1.3	2.1	3.8	5.6	8.1
ImageNet 2011 21K (+1K)	$20,841$ $+1000$	DeViSE	**0.3**	0.8	1.9	3.2	5.3
		ConSE(1)	0.1	1.2	3.0	4.8	7.0
		ConSE(10)	0.2	1.2	3.0	**5.0**	**7.5**
		ConSE(1000)	0.2	1.2	3.0	4.9	7.3

Table 1: Flat hit@k performance of DeViSE [6] and ConSE(T) for $T = 1, 10, 1000$ on ImageNet zero-shot learning task. When testing the methods with the datasets indicated with (+1K), training labels are also included as potential labels within the nearest neighbor classifier, hence the number of candidate labels is 1000 more. In all cases, zero-shot classes did not occur in the training set, and none of the test images is annotated with any of the training labels.

The high level of annotation granularity in Imagenet, *e.g.,* different types of sea lions, creates challenges for recognition systems which are based solely on visual cues. Using models such as ConSE and DeViSE, one can leverage the similarity between the class labels to expand the original predictions of the image classifiers to a list of similar labels, hence better retrieval rates can be achieved.

We report quantitative results in terms of two metrics: "flat" hit@k and "hierarchical" precision@k. Flat hit@k is the percentage of test images for which the model returns the one true label in its top k predictions. Hierarchical precision@k uses the ImageNet category hierarchy to penalize the predictions that are semantically far from the correct labels more than the predictions that are close. Hierarchical precision@k measures, on average, what fraction of the model's top k predictions are among the k most relevant labels for each test image, where the relevance of the labels is measure by their distance in the Imagenet category hierarchy. A more formal definition of hierarchical precision@k is included in the supplementary material of [6]. Hierarchical precision@1 is always equivalent to flat hit@1.

Table 1 shows flat hit@k results for the DeViSE and three versions of the ConSE model. The ConSE model has a hyper-parameter T that controls the number of training labels used for the convex combination of semantic embeddings. We report the results for $T = 1, 10, 1000$ as ConSE(T) in Table 1. Because there are only 1000 training labels, T is bounded by $1 \leq T \leq 1000$. The results are reported on the three test datasets; the dataset difficulty increases from top to bottom in Table 1. For each dataset, we consider including and excluding the training labels within the label candidates used for k-nearest neighbor label ranking (*i.e.,* $\mathcal{Y}_1$ in Eq. (3)). None of the images in the test set are labeled as training labels, so including training labels in the label candidate set for ranking hurts the performance as finding the correct labels is harder in a larger set. Datasets that include training labels in their label candidate set are marked by "(+1K)". The results demonstrate

Test Label Set	Model	Hierarchical precision@k				
		1	2	5	10	20
2-hops	DeViSE	0.06	0.152	0.192	0.217	0.233
	ConSE(10)	**0.094**	**0.214**	**0.247**	**0.269**	**0.284**
2-hops (+1K)	Softmax baseline	0	**0.236**	0.181	0.174	0.179
	DeViSE	**0.008**	0.204	0.196	0.201	0.214
	ConSE(10)	0.003	0.234	**0.254**	**0.260**	**0.271**
3-hops	DeViSE	0.017	0.037	0.191	0.214	0.236
	ConSE(10)	**0.027**	**0.053**	**0.202**	**0.224**	**0.247**
3-hops (+1K)	Softmax baseline	0	0.053	0.157	0.143	0.130
	DeViSE	**0.005**	0.053	0.192	0.201	0.214
	ConSE(10)	0.002	**0.061**	**0.211**	**0.225**	**0.240**
ImageNet 2011 21K	DeViSE	0.008	0.017	0.072	0.085	0.096
	ConSE(10)	**0.014**	**0.025**	**0.078**	**0.092**	**0.104**
ImageNet 2011 21K (+1K)	Softmax baseline	0	0.023	0.071	0.069	0.065
	DeViSE	**0.003**	0.025	0.083	0.092	0.101
	ConSE(10)	0.002	**0.029**	**0.086**	**0.097**	**0.105**

Table 2: Hierarchical precision@k performance of Softmax baseline [7], DeViSE [6], and ConSE(10) on ImageNet zero-shot learning task.

that the ConSE model consistently outperforms the DeViSE on all of the datasets for all values of T. Among different versions of the ConSE, ConSE(10) performs the best. We do not directly compare against the method of Socher et al. [18], but Frome et al. [6] reported that the ranking loss used within the DeViSE significantly outperforms the the squared loss used in [18].

Not surprisingly, the performance of the models is best when training labels are excluded from the label candidate set. All of the models tend to predict training labels more often than test labels, especially at their first few predictions. For example, when training labels are included, the performance of ConSE(10) drops from 9.4% hit@1 to 0.3% on the 2-hops dataset. This suggests that a procedure better than vanilla k-nearest neighbor search needs to be employed in order to distinguish images that do not belong to the training labels. We note that the DeViSE has a slightly lower bias towards training labels as the performance drop after inclusion of training labels is slightly smaller than the performance drop in the ConSE model.

Table 2 shows hierarchical precision@k results for the Softmax baseline, DeViSE, and ConSE(10) on the zero-shot learning task. The results are only reported for ConSE (10) because $T = 10$ seems to perform the best among $T = 1, 10, 1000$. The hierarchical metric also confirms that the ConSE improves upon the DeViSE for zero-shot learning. We did not compare against the Softmax baseline on the flat hit@k measure, because the Softmax model cannot predict any of the test labels. However, using the hierarchical metric, we can now compare with the Softmax baseline when the training labels are also included in the label candidate set (+1K). We find that the top k predictions of the ConSE outperform the Softmax baseline in hierarchical precision@k.

Even though the ConSE model is proposed for zero-shot learning, we assess how the ConSE compares with the DeViSE and the Softmax baseline on the standard classification task with the training 1000 labels, *i.e.,* the training and test labels are the same. Table 3 and 4 show the flat hit@k and hierarchical precision@k rates on the 1000-class learning task. According to Table 3, the ConSE(10) model improves upon the Softmax baseline in hierarchical precision at 5, 10, and 20, suggesting that the mistakes made by the ConSE model are on average more semantically consistent with the correct class labels, than the Softmax baseline. This improvement is due to the use of label embedding vectors learned from Wikipedia articles. However, on the 1000-class learning task, the ConSE(10) model underperforms the DeViSE model. We note that the DeViSE model is trained with respect to a k-nearest neighbor retrieval objective on the same specific set of 1000 labels, so its better performance on this task is expected.

Although the DeViSE model performs better than the ConSE on the original 1000-class learning task (Table 3, 4), it does not generalize as well as the ConSE model to the unseen zero-shot learning categories (Table 1, 2). Based on this observation, we conclude that a better k-nearest neighbor

Test Label Set	Model	Hierarchical precision@k				
		1	2	5	10	20
ImageNet 2011 1K	Softmax baseline	**0.556**	**0.452**	0.342	0.313	0.319
	DeViSE	0.532	0.447	**0.352**	**0.331**	**0.341**
	ConSE (1)	0.551	0.422	0.32	0.297	0.313
	ConSE (10)	0.543	0.447	0.348	0.322	0.337
	ConSE (1000)	0.539	0.442	0.344	0.319	0.335

Table 3: Hierarchical precision@k performance of Softmax baseline [7], DeViSE [6], and ConSE on ImageNet original 1000-class learning task.

Test Label Set	Model	Flat hit@k (%)			
		1	2	5	10
ImageNet 2011 1K	Softmax baseline	**55.6**	**67.4**	**78.5**	**85.0**
	DeViSE	53.2	65.2	76.7	83.3
	ConSE (1)	55.1	57.7	60.9	63.5
	ConSE (10)	54.3	61.9	68.0	71.6
	ConSE (1000)	53.9	61.1	67.0	70.6

Table 4: Flat hit@k performance of Softmax baseline [7], DeViSE [6], and ConSE on ImageNet original 1000-class learning task.

classification on the training labels, does not automatically translate into a better k-nearest neighbor classification on a zero-shot learning task. We believe that the DeViSE model suffers from a variant of overfitting, which is the model has learned a highly non-linear and complex embedding function for images. This complex embedding function is well suited for predicting the training label embeddings, but it does not generalize well to novel unseen label embedding vectors. In contrast, a simpler embedding model based on convex combination of semantic embeddings (ConSE) generalizes more reliably to unseen zero-shot classes, with little chance of overfitting.

Implementation details. The ConSE(1) model takes the top-1 prediction of the convolutional net, and expands it to a list of labels based on the similarity of the label embedding vectors. To implement ConSE(1) efficiently, one can pre-compute a list of test labels for each training label, and simply predict the corresponding list based on the top prediction of the convolutional net. The top prediction of the ConSE(1) occasionally differs from the top prediction of the Softmax baseline due to a detail of our implementation. In the Imagenet experiments, following the setup of the DeViSE model, there is not a one-to-one correspondence between the class labels and the word embedding vectors. Rather, because of the way the Imagenet synsets are defined, each class label is associated with several synonym terms, and hence several word embedding vectors. In the process of mapping the Softmax scores to an embedding vector, the ConSE model first averages the word vectors associated with each class label, and then linearly combine the average vectors according to the Softmax scores. However, when we rank the word vectors to find the k most likely class labels, we search over individual word vectors, without any averaging of the synonym words. Thus, the ConSE(1) might produce an average embedding which is not the closest vector to any of the word vectors corresponding to the original class label, and this results in a slight difference in the hit@1 scores for ConSE(1) and the Softmax baseline. While other alternatives exist for this part of the algorithm, we intentionally kept the ranking procedure exactly the same as the DeViSE model to perform a direct comparison.

6 Conclusion

The ConSE approach to mapping images into a semantic embedding space is deceptively simple. Treating classifier scores as weights in a convex combination of word vectors is perhaps the most direct method imaginable for recasting an n-way image classification system as image embedding system. Yet this method outperforms more elaborate joint training approaches both on zero-short learning and on performance metrics which weight errors based on semantic quality. The success of this method undoubtedly lays is its ability to leverage the strengths inherent in the state-of-the-art image classifier and the state-of-the-art text embedding system from which it was constructed.

While it draws from their strengths, we have no reason to believe that ConSE depends on the details the visual and text models from which it is constructed. In particular, though we used a deep convolutional network with a Softmax classifier to generate the weights for our linear combination, any visual object classification system which produces relative scores over a set of classes is compatible with the ConSE framework. Similarly, though we used semantic embedding vectors which were the side product of an unsupervised natural language processing task, the ConSE framework is applicable to other alternative representation of text in which similar concepts are nearby in vector space. The choice of the training corpus for the word embeddings affects the results too.

One feature of the ConSE model which we did not exploit in our experiments is its natural representation of confidence. The norm of the vector that ConSE assigns to an image is a implicit expression of the model's confidence in the embedding of that image. Label assignments about which the Softmax classifier is uncertain be given lower scores, which naturally reduces the magnitude of the ConSE linear combination, particularly if Softmax probabilities are used as weights without renormalization. Moreover, linear combinations of labels with disparate semantics under the text model will have a lower magnitude than linear combinations of the same number of closely related labels. These two effects combine such that ConSE only produces embeddings with an L2-norm near 1.0 for images which were either nearly completely unambiguous under the image model or which were assigned a small number of nearly synonymous text labels. We believe that this property could be fruitfully exploited in settings where confidence is a useful signal.

References

[1] E. Bart and S. Ullman. Cross-generalization: learning novel classes from a single example by feature replacement. CVPR, 2005.

[2] Y. Bengio, H. Schwenk, J.-S. Senécal, F. Morin, and J.-L. Gauvain. Neural probabilistic language models. Innovations in Machine Learning, 2006.

[3] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. CVPR, 2009.

[4] A. Farhadi, I. Endres, D. Hoiem, and D. Forsyth. Describing objects by their attributes. CVPR, 2009.

[5] L. Fei-Fei, R. Fergus, and P. Perona. One-shot learning of object categories. *IEEE Trans. PAMI*, 28:594–611, 2006.

[6] A. Frome, G. S. Corrado, J. Shlens, S. Bengio, J. Dean, M. Ranzato, and T. Mikolov. Devise: A deep visual-semantic embedding model. NIPS, 2013.

[7] A. Krizhevsky, I. Sutskever, and G. Hinton. Imagenet classification with deep convolutional neural networks. NIPS, 2012.

[8] B. M. Lake, R. Salakhutdinov, J. Gross, and J. B. Tenenbaum. One shot learning of simple visual concepts. CogSci, 2011.

[9] C. H. Lampert, H. Nickisch, and S. Harmeling. Learning to detect unseen object classes by between-class attribute transfer. CVPR, 2009.

[10] H. Larochelle, D. Erhan, and Y. Bengio. Zero-data learning of new tasks. AAAI, 2008.

[11] T. Mensink, J. Verbeek, F. Perronnin, and G. Csurka. Metric learning for large scale image classification: Generalizing to new classes at near-zero cost. ECCV, 2012.

[12] T. Mikolov, K. Chen, G. Corrado, and J. Dean. Efficient estimation of word representations in vector space. ICLR, 2013.

[13] E. G. Miller, N. E. Matsakis, and P. A. Viola. Learning from one example through shared densities on transforms. CVPR, 2000.

[14] M. Palatucci, D. Pomerleau, G. E. Hinton, and T. M. Mitchell. Zero-shot learning with semantic output codes. NIPS, 2009.

[15] D. Parikh and K. Grauman. Relative attributes. ICCV, 2011.

[16] M. Rohrbach, M. Stark, and B. Schiele. Evaluating knowledge transfer and zero-shot learning in a large-scale setting. CVPR, 2011.

[17] M. Rohrbach, M. Stark, and B. Schiele. Evaluating knowledge transfer and zero-shot learning in a large-scale setting. CVPR, 2011.

[18] R. Socher, M. Ganjoo, H. Sridhar, O. Bastani, C. D. Manning, and A. Y. Ng. Zero-shot learning through cross-modal transfer. NIPS, 2013.

[19] J. Weston, S. Bengio, and N. Usunier. Wsabie: Scaling up to large vocabulary image annotation. IJCAI, 2011.