A Probabilistic Model for Joint Learning of Word Embeddings from Texts and Images

Melissa Ailem[1,2], **Bowen Zhang**[2], **Aurelien Bellet**[1], **Pascal Denis**[1] **and Fei Sha**[2,3]
[1] INRIA, France
[2] University of Southern California, Los Angeles, CA
[1] {melissa.ailem,aurelien.bellet,pascal.denis}@inria.fr
[2] {zhan734, ailem, feisha}@usc.edu [3] fsha@netflix.com*

Abstract

Several recent studies have shown the benefits of combining language and perception to infer word embeddings. These multimodal approaches either simply combine pre-trained textual and visual representations (e.g. features extracted from convolutional neural networks), or use the latter to bias the learning of textual word embeddings. In this work, we propose a novel probabilistic model to formalize how linguistic and perceptual inputs can work in concert to explain the observed word-context pairs in a text corpus. Our approach learns textual and visual representations jointly: latent visual factors couple together a skip-gram model for co-occurrence in linguistic data and a generative latent variable model for visual data. Extensive experimental studies validate the proposed model. Concretely, on the tasks of assessing pairwise word similarity and image/caption retrieval, our approach attains equally competitive or stronger results when compared to other state-of-the-art multimodal models.

1 Introduction

Continuous-valued vector representation of words has been one of the key components in neural architectures for natural language processing (Mikolov et al., 2013; Pennington et al., 2014; Levy and Goldberg, 2014). The main idea is based on the distributional hypothesis (Harris, 1954), which states that words used in similar contexts have similar semantic meanings. To this end, words are mapped to points in an Euclidean space such that the displacement between their coordinates (i.e., embeddings) reflects similarity and difference in semantics (Pennington et al., 2014). As such, word embeddings have been shown to be useful in determining semantic and syntactic similarity between individual words (Mikolov et al., 2013; Baroni et al., 2014; Levy et al., 2015), as well as in downstream NLP tasks, e.g., sentiment analysis, question answering, and coreference resolution, just to name a few.

Most existing approaches rely solely on text corpora to infer word representations. While successful, the embeddings produced by such models do not necessarily reflect all inherent aspects of human semantic knowledge, such as the perceptual aspect (Feng and Lapata, 2010). This has motivated many researchers to explore different ways to infuse visual information, often represented in the form of pre-computed visual features, into word embeddings (Kiela and Bottou, 2014; Silberer et al., 2017; Collell et al., 2017; Lazaridou et al., 2015). The main theme is to take either the text embeddings, or the visual features or both *as such* to derive multimodal embeddings: through concatenation (Kiela and Bottou, 2014), or by treating visual features as regression targets (Lazaridou et al., 2015; Collell et al., 2017).

Despite the success of these prior efforts in yielding multimodal embeddings and applying them to downstream NLP tasks, there are still several deficiencies. In particular, the visual features (as such) are not guaranteed to be suitable for the word embedding task since they are typically optimized independently for another objective (e.g., image classification). Hence, fusing *pre-computed* word representations and visual features may not be a good strategy.

To address the above issues, we explore a new way to integrate linguistic and perceptual information. We develop a new model which jointly learns word embeddings from text and extracts latent visual information, from pre-computed visual features, that could supplement the linguistic embeddings in modeling the co-occurrence of words

*On leave from U of Southern California

and their contexts in a corpus. Instead of using pre-trained visual features as it is or as regression targets, we posit that they contain latent perceptual information that could complement text in representing words.

More specifically, the proposed model consists of two components. The visual component is an unsupervised probabilistic model for learning latent factors that generates the visual data. The linguistic component is a revised SKIP-GRAM model in which the text embeddings work *in concert* with the latent visual factors to explain the occurrence of word-context pairs in a corpus. One advantage of our joint modeling is that it allows two-way interaction. On one hand, the linguistic information can guide the extraction of latent visual factors. On the other hand, the extracted visual factors can improve the modeling of word-context co-occurrences in text data. Another appealing property of our model is its natural ability to propagate perceptual information to the embeddings of words lacking visual features (e.g., abstract words) during learning.

We conduct extensive quantitative and qualitative experiments to examine and understand the effectiveness of our approach, on the tasks of word similarity and image/caption retrieval. We show its matching or stronger performance when compared to other state-of-the-art approaches for learning multimodal embeddings.

2 Our Approach

We start by introducing the problem setup and notations. We then describe our model, namely PIXIE (ProbabIlistic teXtual Image Embeddings), for joint learning of word representations from text and perceptual information.

2.1 Setup and Background

We are given a corpus of H tokens (words): $w_1, \ldots, w_i, \ldots, w_H$. From the corpus, we form a collection of word-context pairs $\delta_{w,c} = (w, c)$, such that $w \in V_w$, $c \in V_c$, with V_w and V_c denoting respectively the word and context vocabularies. As in most previous work, the contexts for word w_i are the words that surround it in a L-sized window. We introduce the binary indicator variables y_{wc}, such that $y_{wc} = 1$ if $\delta_{w,c}$ appears in our collection, and $y_{wc} = 0$ otherwise.

For some words with visual grounding (we will refer to them as *visual words*), we have access to a visual representation $\boldsymbol{x}_w$. In practice, we use convolutional net features (see Section 4 for details).

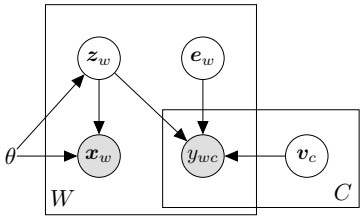

Figure 1: Plate representation of our model PIXIE. The model consists of a generative model for visual data and a conditional model for text data. The latent visual factors $\boldsymbol{z}$ and text embedding $\boldsymbol{e}$ jointly predict the word-context pair's label y.

SKIP-GRAM WITH NEGATIVE SAMPLING (SGNS) The SGNS's objective is to learn word representations that are good at distinguishing the observed pairs ($y_{wc} = 1$) from non-observed or "negative" pairs ($y_{wc} = 0$), using logistic regression. Formally, SGNS maximizes the following log-likelihood:

$$\sum_{w,c} [y_{wc} \log \sigma(\boldsymbol{v}_c^\mathsf{T} \boldsymbol{e}_w) + (1 - y_{wc}) \log \sigma(-\boldsymbol{v}_c^\mathsf{T} \boldsymbol{e}_w)], \quad (1)$$

where $\sigma(\cdot)$ is the sigmoid function, $\boldsymbol{v}_c$, $\boldsymbol{e}_w$ denote respectively the vectors for the context c and target word w. The second term in (1) is intractable due to the large number of possible negative pairs, and is approximated by sampling N negative examples $\{c'_i\}_{i=1}^N$ for every observed pair of words and their contexts. This gives rise to the following objective function for each observed pair:

$$\log \sigma(\boldsymbol{v}_c^\mathsf{T} \boldsymbol{e}_w) + \sum_{i=1}^{N} \log \sigma(-\boldsymbol{v}_{c'_i}^\mathsf{T} \boldsymbol{e}_w), \quad (2)$$

where c'_i is a (negative) context that does not appear in the context of w (Mikolov et al., 2013). In practice, criterion (2) is optimized in an online fashion, by using Stochastic Gradient Descent (SGD) over the observed pairs δ_{wc} in the corpus. Each observed pair δ_{wc} typically occurs several times in the corpus, therefore performing SGD over the corpus amounts to weighting equation (2) by the number of occurrence of each pair.

2.2 Joint Visual and Text Modeling

We now describe our model, namely PIXIE (ProbabIlistic teXtual Image Embeddings) illustrated in Fig. 1, for joint learning of word representations from textual and perceptual information.

Formally, PIXIE is a probabilistic model of image features $\boldsymbol{x}$ and word-context pairs' labels $\boldsymbol{y}$. Similar to SKIP-GRAM, PIXIE represents each word w and context c with low dimensional embeddings noted respectively $\boldsymbol{e}_w \in \mathbb{R}^K$ and $\boldsymbol{v}_c \in \mathbb{R}^K$. PIXIE further assumes latent visual factors, $\boldsymbol{z}_w \in \mathbb{R}^K$, for each word's visual representation $\boldsymbol{x}_w$. Next we describe the two main components of PIXIE, namely *textual* and *perceptual*, in more details.

Perceptual Component Each visual vector $\boldsymbol{x}_w$ is drawn conditional on its latent representation $\boldsymbol{z}_w$, i.e., $\boldsymbol{x}_w \sim p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z}_w)$, with $p_{\boldsymbol{\theta}}(\boldsymbol{z}) = \mathcal{N}(0, \mathbf{I})$. Since $\boldsymbol{x}_w$ is real valued, we let $p_{\boldsymbol{\theta}}(\boldsymbol{x}_w|\boldsymbol{z}_w)$ be a Gaussian parameterized by a *generative neural network* (or decoder). That is,

$$p_{\boldsymbol{\theta}}(\boldsymbol{x}_w|\boldsymbol{z}_w) = \mathcal{N}(\boldsymbol{x}_w|\boldsymbol{\mu}_{\boldsymbol{\theta}}(\boldsymbol{z}_w), \boldsymbol{\Sigma}_{\boldsymbol{\theta}}(\boldsymbol{z}_w)). \quad (3)$$

For tractability purposes, $\boldsymbol{\Sigma}_{\boldsymbol{\theta}}$ is restricted to be diagonal. Moreover, both the co-variance $\boldsymbol{\Sigma}_{\boldsymbol{\theta}}(\boldsymbol{z}_w)$ (its diagonal) and the mean $\boldsymbol{\mu}_{\boldsymbol{\theta}}$ are the outputs of a decoder network with parameters $\boldsymbol{\theta}$ and input $\boldsymbol{z}_w$.

Textual Component To model the occurrence/absence of word-context pair δ_{wc} in the linguistic corpus, we adopt a Bernoulli (Ber) model:

$$p(y_{wc}|\, \boldsymbol{e}_w, \boldsymbol{v}_c, \boldsymbol{z}_w) = \text{Ber}(\sigma[f(\boldsymbol{e}_w, \boldsymbol{v}_c, \boldsymbol{z}_w)]) \quad (4)$$

The function $f(\cdot)$ defines how multimodal embeddings are fused. While many choices can be experimented, we use the simple additive model:

$$f(\boldsymbol{e}_w, \boldsymbol{v}_c, \boldsymbol{z}_w) = \boldsymbol{v}_c^\mathsf{T}(\boldsymbol{e}_w + \boldsymbol{z}_w). \quad (5)$$

For words without visual representation, we simply set the corresponding latent factors $\boldsymbol{z}_w$ to the zero vector. Note that, without the visual factors $\boldsymbol{z}_w$, equation (4) reduces to the Skip-Gram with negative sampling objective (1).

Joint Model The perceptual and the textual information interact through the shared latent $\boldsymbol{z}_w$. The joint model of the above two sources of information takes the following form:

$$p(\boldsymbol{x}_w, y_{wc}|\, \boldsymbol{e}_w, \boldsymbol{v}_c) = \int p_{\boldsymbol{\theta}}(\boldsymbol{z}_w) p_{\boldsymbol{\theta}}(\boldsymbol{x}_w|\boldsymbol{z}_w) p(y_{wc}|\, \boldsymbol{e}_w, \boldsymbol{v}_c, \boldsymbol{z}_w) d\boldsymbol{z}_w \quad (6)$$

The intuition behind our joint formulation is to let the textual information guide the extraction of latent visual factors $\boldsymbol{z}_w$. Through equations (4) and (5), the model will put high probability on factors $\boldsymbol{z}_w$ reflecting patterns that can supplement the linguistic embeddings $\boldsymbol{e}_w$ in explaining the word-context co-occurrences. Thus, the extracted latent visual factors can contribute to improve the performance on predicting the occurrence of a word and its contexts in the linguistic corpus, which would encourage the model to leverage the perceptual information. The underlying assumption here is to infer visual and textual embeddings that can work in concert to represent words.

Visual Information Propagation Equation (5) implies that the embeddings $\boldsymbol{z}$, $\boldsymbol{e}$ and $\boldsymbol{v}$ will affect each other during the learning process. Interestingly, if a non-visual word w_1 shares a similar context c with a visual word w_2, then the factor $\boldsymbol{z}_{w_2}$ will affect $\boldsymbol{e}_{w_1}$ via $\boldsymbol{v}_c$. In other words, our formulation makes it possible to implicitly propagate perceptual information from one word to another through shared contexts. We illustrate this aspect in our experiments.

2.3 Approximate Inference and Learning

Training PIXIE amounts to inferring the posterior over the visual latent factors, $p_{\boldsymbol{\theta}}(\boldsymbol{z}|\boldsymbol{x}, y)$, as well as finding the decoder's parameters $\boldsymbol{\theta}$, the word and context embeddings, $\boldsymbol{e}$ and $\boldsymbol{v}$, that maximize the likelihood (6). However, as in many complex probabilistic models, the likelihood (due to the integral over $\boldsymbol{z}$) and the posterior are intractable. We therefore resort to approximation techniques. More precisely, we rely on Variational Inference (VI) (Blei et al., 2017). The idea of VI is to introduce a tractable approximate posterior distribution $q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x})$ (the variational distribution) and optimize a lower bound on the likelihood, known as Evidence Lower BOund (ELBO). The latter can be written for each word w as follows:

$$\mathcal{L}_w = \mathbb{E}_q[\log p_{\boldsymbol{\theta}}(\boldsymbol{x}_w|\boldsymbol{z}_w) + \sum_c \log p(y_{wc}|\boldsymbol{e}_w, \boldsymbol{v}_c, \boldsymbol{z}_w)] - \text{KL}(q_{\boldsymbol{\phi}}(\boldsymbol{z}_w|\boldsymbol{x}_w)\|p_{\boldsymbol{\theta}}(\boldsymbol{z}_w)) \quad (7)$$

where $\text{KL}(\cdot\|\cdot)$ is the Kullback-Leibler divergence. The variational distribution is chosen to be a multivariate Gaussian parameterized by an *inference network* (or encoder) which takes $\boldsymbol{x}$ as input, namely

$$q_{\boldsymbol{\phi}}(\boldsymbol{z}_w|\boldsymbol{x}_w) = \mathcal{N}(\boldsymbol{z}_w|\boldsymbol{\mu}_{\boldsymbol{\phi}}(\boldsymbol{x}_w), \boldsymbol{\Sigma}_{\boldsymbol{\phi}}(\boldsymbol{x}_w)), \quad (8)$$

where we drop the dependency on all y_{wc} variables to be computationally tractable. The pair of encoder and decoder neural networks gives rise to the

interpretation of PIXIE's visual component (formed by $\boldsymbol{z}$ and $\boldsymbol{x}$) as a probabilistic autoencoder. In fact, if we drop the textual part in PIXIE, namely y, $\boldsymbol{e}$ and $\boldsymbol{v}$, then we recover the Variational Auto-Encoder (VAE) (Kingma and Welling, 2013).

Lastly, we approximate the intractable (i) expectation with respect to $q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x})$ and the (ii) sum over the negative pairs in (7), by relying on a Monte Carlo estimator of $\mathcal{L}$. Concretely, for (ii) we use negative sampling as in (2). Concerning (i), for every observed $\boldsymbol{x}_w$, we sample $\{\boldsymbol{z}_w^{(j)}\}_{j=1}^J$ from $q_{\boldsymbol{\phi}}(\boldsymbol{z}_w|\boldsymbol{x}_w)$ using the *reparameterization trick* (Kingma and Welling, 2013), i.e., $\boldsymbol{z}_w^{(j)} = \boldsymbol{\mu}_{\boldsymbol{\phi}}(\boldsymbol{x}_w) + \boldsymbol{\Sigma}_{\boldsymbol{\phi}}(\boldsymbol{x}_w)\boldsymbol{\epsilon}$, with $\boldsymbol{\epsilon} \sim \mathcal{N}(0, \mathbf{I})$. Then we approximate $\mathcal{L}$ with:

$$\begin{aligned}
&\widetilde{\mathcal{L}}_w = \\
&\tfrac{1}{J}\sum_j \log p_{\boldsymbol{\theta}}(\boldsymbol{x}_w|\boldsymbol{z}_w^{(j)}) - \mathrm{KL}(q_{\boldsymbol{\phi}}(\boldsymbol{z}_w|\boldsymbol{x}_w)\|p_{\boldsymbol{\theta}}(\boldsymbol{z}_w)) \\
&+ \tfrac{1}{J}\sum_{c,j} y_{wc} \log \sigma[\boldsymbol{v}_c^\mathsf{T}(\boldsymbol{e}_w + \boldsymbol{z}_w^{(j)})] \\
&+ \tfrac{1}{J}\sum_{c,j}\sum_{i=1}^N y_{wc} \log(\sigma[-\boldsymbol{v}_{c_i'}^\mathsf{T}(\boldsymbol{e}_w + \boldsymbol{z}_w^{(j)})]). \qquad (9)
\end{aligned}$$

The last two summands correspond to the familiar conditional likelihood term in the SGNS model, augmented with latent visual factors.

We optimize the objective (9) via SGD with respect to both the encoder/decoder networks parameters ($\boldsymbol{\theta}$ and $\boldsymbol{\phi}$) and the embeddings ($\boldsymbol{e}$ and $\boldsymbol{v}$). We evaluate the gradients of $\widetilde{\mathcal{L}}$, with respect to $\boldsymbol{\theta}$ and $\boldsymbol{\phi}$ using backpropagation. Similarly, the gradient with respect to $\boldsymbol{e}$ and $\boldsymbol{v}$ can be easily carried out using automatic differentiation tools. Our learning procedure is summarized in Algorithm 1.

Algorithm 1 Variational PIXIE

Input: $\boldsymbol{x}$, y, sample sizes B and J
Steps:
Randomly initialize $\boldsymbol{\theta}$, $\boldsymbol{\phi}$, $\boldsymbol{e}$ and $\boldsymbol{v}$
repeat
- Draw a minibatch W^B of words: $w^{(1)}, \ldots, w^{(B)}$
- For each $\boldsymbol{x}_w$ with $w \in W^B$, sample $\{\boldsymbol{z}_w^{(j)}\}_{j=1}^J$ from $q_{\boldsymbol{\phi}}(\boldsymbol{z}_w|\boldsymbol{x}_w)$ using the *reparameterization trick*. For each observed pair δ_{wc} draw N negative examples $\{c_i'\}_{i=1}^N$.
- Compute the estimator $\widetilde{\mathcal{L}}_{W^B} \leftarrow \sum_{w \in W^B} \widetilde{\mathcal{L}}_w$
- Compute the gradient: $\mathbf{G} \leftarrow \nabla_{\boldsymbol{\theta},\boldsymbol{\phi},\boldsymbol{e},\boldsymbol{v}} \widetilde{\mathcal{L}}_{W^B}$
- Use $\mathbf{G}$ to update $\boldsymbol{\theta}$, $\boldsymbol{\phi}$, $\boldsymbol{e}$ and $\boldsymbol{v}$ (e.g., with ADAM)

until convergence
return $\boldsymbol{\theta}$, $\boldsymbol{\phi}$, $\boldsymbol{e}$ and $\boldsymbol{v}$

Inference Once the parameters of the model are learned, for any given word with or without visual representation, we can compute its multimodal embedding. As a short hand, let the binary variable m_w denote whether or not the word w has a visual representation. The multimodal embedding for w can be written as

$$\boldsymbol{s}_w = \boldsymbol{e}_w + m_w \boldsymbol{\mu}(\boldsymbol{x}_w), \qquad (10)$$

where $\boldsymbol{\mu}(\boldsymbol{x}_w) = \mathbb{E}_q(\boldsymbol{z}_w)$ is the output of the encoding neural network, cf. Eq. (8).

In our experiments, we have also studied an alternative way to compose multimodal embeddings by concatenating the two vectors $\boldsymbol{e}_w$ and $\boldsymbol{\mu}(\boldsymbol{x}_w)$

$$\boldsymbol{t}_w = [\boldsymbol{e}_w \;\; m_w \boldsymbol{\mu}(\boldsymbol{x}_w)]. \qquad (11)$$

Note that for non-visual words, only zeros are appended to $\boldsymbol{e}_w$. One advantage of $\boldsymbol{t}$ over $\boldsymbol{s}$ is that if one uses distances to measure similarity, $\boldsymbol{t}$ can be seen as a simple summation of distances in two different spaces (in terms of $\boldsymbol{e}$ and $\boldsymbol{\mu}$ respectively).

3 Related Work

Combining language and perception has been recently considered in various NLP tasks such as machine translation (Calixto and Liu, 2017), visual question generation (Mostafazadeh et al., 2016), image captioning (Klein et al., 2015), etc. In this work, we focus on learning word embeddings from images and texts.

Multimodal embeddings have been studied in several recent research work. One strategy is to obtain word embeddings from linguistic data and visual data *independently* and then proceed with some kind of fusion steps. Kiela and Bottou (2014) simply concatenates pre-trained linguistic word embeddings and visual features computed by convolutional nets. Bruni et al. (2014) performs an additional step of dimensionality reduction via singular value decomposition. Silberer et al. (2017) extend on this work by feeding the linguistic embedding and visual features into a stacked auto-encoder for nonlinear dimensionality reduction. The above-mentioned approaches perform a two-stage process to derive multimodal representations (unimodal inference followed by fusion) and have been evaluated only on words for which both perceptual and textual representations are available.

A standing question is how to propagate visual information from words with visual features to words lacking them (for instance, abstract words). While the previous methods fall short on that, the recent work by Collell et al. (2017) addresses this challenge by learning a mapping from language to

vision, using a set of words with known linguistic embeddings and visual features. This mapping can then be used to infer visual representations for new words from their textual embeddings.

All the aforementioned methods rely on independently pre-trained linguistic embeddings and visual features. In this work, we propose a different strategy, which consists in adapting those representations so that the information can be fused in earlier stages. In this respect, the closest work to ours is (Lazaridou et al., 2015), which proposes to augment the SKIP-GRAM objective function with a term mapping the textual embeddings to the visual features. Crudely, the linguistic embeddings must therefore predict both the text co-occurrences and (pre-trained) visual features. We emphasize two key differences with our approach. First, instead of performing a regression or mapping from the textual embeddings to the visual features, our model learns to infer perceptual latent factors to retain only the portion of visual information that can supplement the linguistic embeddings in representing words. Second, while Lazaridou et al. (2015) combines two objectives, we use a joint probabilistic model integrating both visual and text information in a principled way. Specifically, our model seeks latent factors that are good at explaining the word-context co-occurrences. For instance, a visual feature of (an image of) OCEAN often contains information about SKY and BLUE — such visual information could be beneficial to predict co-occurrence of tokens in the context of OCEAN. This desiderata further strengthens the learned embeddings to be visually grounded. In our experiments, we show that our approach tends to group concrete visually similar concepts together.

4 Experiments

In this section, we evaluate our model and contrast it to other competing approaches on two different tasks: word similarity and image/caption retrieval.

4.1 Setup

Text corpus We use the Text8 WIKIPEDIA corpus[1] containing over 17 million tokens. Text8 was pre-processed to contain only letters and non-consecutive spaces. After removing infrequent words, we obtain a vocabulary of $50,000$ unique words.

Image features We use the ImageNet dataset (Russakovsky et al., 2015), including the fall ImageNet 2011 release (Deng et al., 2009). It contains $14,188,125$ images organized according to $21,842$ synsets of WordNet (Fellbaum, 1998). Each synset contains 600 images on average. To extract image features, we rely on the Caffe toolkit (Jia et al., 2014) and use the GoogLeNet convolutional neural nets (Szegedy et al., 2015) pre-trained on the 1000 synsets of ILSVRC 2012. The 1024-dimensional activation of the pooling units (before the softmax layer) are then taken as our image features.

Visual representation of words For each word in the vocabulary, we recover all the synsets that it belongs to using the WordNet interface of the NLTK module (Python) (Bird et al., 2009). We then remove the synsets not covered by our ImageNet dataset. This results in 9,713 words, out of the 50,000 words in the vocabulary. For each visual word, we randomly draw 1,000 distinct images in ImageNet. If the number of images for a word is less than 1,000, we increase the coverage using images belonging to the hypernyms of the considered word's synsets, as in (Kiela and Bottou, 2014). We then take the average of these features as the word's visual representation $\boldsymbol{x}$.

Hyper-parameter setting For all models, we set the dimension of linguistic and visual embeddings, $\boldsymbol{e}$ and $\boldsymbol{z}$, to 100, following many previous works. In our model, the encoder/decoder neural networks are implemented as one-hidden-layer neural nets with 500 hidden units each. The dimensions of the inputs and the outputs of the decoder neural networks are 100 and $1,024$ respectively ($1,024$ and 100 for the encoder). For the encoder, the hidden units are hyperbolic tangent, and the output units are linear. For the decoder, the hidden units are hyperbolic while the outputs are sigmoid. For SGNS, we set the window size L to 10 and the number of negative samples to 64. Our model is learned by Stochastic Gradient Descent using the ADAM optimizer (Kingma and Ba, 2014) with a learning rate set to 0.001.

Table 1: Datasets for the task of word similarity.

Datasets	#word pairs
MEN (Bruni et al., 2014)	3000
EN-MC (Miller and Charles, 1991)	31
EN-RG (Rubenstein and Goodenough, 1965)	65
SimLex (Hill et al., 2015)	999
MTurk (Radinsky et al., 2011; Halawi et al., 2012)	287
WORDSIM (Finkelstein et al., 2001)	350
REL (Agirre et al., 2009)	150
SIM (Agirre et al., 2009)	200
SEMSIM (Silberer and Lapata, 2014)	5494
VISSIM (Silberer and Lapata, 2014)	5494

[1] http://mattmahoney.net/dc/textdata

Table 2: Results on word similarity task. Reported are the Spearman's rank order correlation between model prediction and human judgment (higher is better and bolds highlight the best methods). See text for details.

Models	Semantic/taxonomic similarity										General relatedness								Visual similarity		REL+SIM	
	SEMSIM		SimLex		SIM		EN-RG		EN-MC		MEN		REL		MTurk				VISSIM		WORDSIM	
	100%	98%	100%	39%	100%	44%	100%	72%	100%	73%	100%	54%	100%	53%	100%	26%			100%	98%	100%	39%
CNN	-	0.49	-	0.41	-	0.49	-	0.54	-	0.46	-	0.54	-	0.20	-	0.18			-	0.53	-	0.28
VAE	-	0.65	-	0.43	-	0.51	-	0.56	-	0.55	-	0.62	-	0.22	-	0.40			-	0.62	-	0.37
SGNS	0.50	0.50	0.33	0.35	0.66	0.66	0.60	0.55	0.60	0.52	0.65	0.67	**0.56**	0.51	**0.65**	0.63			0.38	0.38	**0.61**	0.60
CNN⊕SGNS†	-	0.67	-	0.48	-	0.65	-	0.60	-	0.55	-	0.74	-	0.44	-	0.51			-	**0.63**	-	0.56
VAE⊕SGNS	-	0.70	-	0.51	-	0.67	-	0.61	-	0.60	-	**0.76**	-	0.45	-	0.55			-	**0.63**	-	0.56
V-SGNS‡	0.58	0.58	0.29	0.30	0.66	0.71	**0.73**	**0.73**	**0.69**	0.69	0.64	0.65	0.51	0.52	0.60	0.65			0.42	0.42	0.59	0.64
IV-SGNS§ (LINEAR)	0.49	0.50	0.31	0.33	0.55	0.61	0.58	0.56	0.59	0.65	0.60	0.62	0.41	0.38	0.57	**0.71**			0.36	0.37	0.46	0.51
IV-SGNS§ (NONLINEAR)	0.44	0.44	0.30	0.32	0.53	0.59	0.54	0.53	0.59	0.63	0.57	0.59	0.40	0.37	0.56	**0.71**			0.32	0.33	0.44	0.48
PIXIE$_+$	0.63	0.63	0.35	0.48	0.63	**0.72**	0.65	0.60	0.62	0.62	0.64	0.73	0.46	0.56	0.55	0.55			0.54	0.54	0.50	0.59
PIXIE$_\oplus$	**0.71**	**0.71**	**0.39**	**0.53**	**0.68**	0.71	**0.73**	**0.73**	**0.69**	**0.71**	**0.68**	**0.76**	0.52	**0.59**	0.60	0.59			0.60	0.61	0.58	**0.65**

†:(Kiela and Bottou, 2014), ‡: (Lazaridou et al., 2015), §: (Collell et al., 2017)

4.2 Task 1: Word Similarity

Datasets Word similarity is a common type of evaluation task for measuring the effectiveness of word embeddings. To this end, we retain 10 benchmark datasets consisting of pairs of words associated with similarity scores given by human judges. Table 1 summarizes their basic properties. There are different types of similarities being assessed. SEMSIM, SimLex, SIM, EN-RG and EN-MC focus on *semantic or taxonomic similarity* — e.g. CAR is similar to AUTOMOBILE. MEN, REL and MTurk consider *general relatedness* — e.g. CAR is related to GARAGE. VISSIM is about *visual similarity* — e.g. GOOSE looks like SWAN. Note that SIM and REL are the similarity and relatedness subsets of the full WORDSIM dataset (Finkelstein et al., 2001) respectively. VISSIM contains the same word pairs as SEMSIM.

Competing models We benchmark our model PIXIE against several strong uni- and multi-modal models listed below:

- SGNS: Skip-Gram with Negative Sampling (Mikolov et al., 2013). Without the visual component, PIXIE reduces to SGNS. We can thus assess the impact of the perceptual information by comparing PIXIE to SGNS.
- VAE: Vatiational Auto-Encoder (Kingma and Welling, 2013), which corresponds to the visual-specific component of PIXIE.
- CNN: Visual features extracted from a convolutional neural net as described in Section 4.1.
- CNN⊕SGNS (Kiela and Bottou, 2014): Concatenation of CNN and SKIP-GRAM embeddings.
- VAE⊕SGNS: Concatenation of VAE and SKIP-GRAM embeddings.
- V-SGNS (Lazaridou et al., 2015): A multimodal approach which augments SGNS with a term that treats CNN visual features as regression targets. Comparisons with V-SGNS will allow us to evaluate the impact of our modeling assumptions.
- IV-SGNS (Collell et al., 2017): Learns a mapping from SGNS embeddings to CNN visual features.

Due to a large degree of discrepancies in experimental setups across previously published methods and results,[2] we re-implemented all the baselines and evaluate them under the same conditions.[3] For (Lazaridou et al., 2015), we implemented its model "A" as model "B" is comparable according to the original authors. For (Collell et al., 2017), we implemented both linear and nonlinear variants.

Evaluation metrics We use the cosine to measure the similarity between word representations. To assess the coherence between human ratings and models' predictions, we use the Spearman correlation coefficient.

4.2.1 Main results

The results across different datasets are shown in Table 2. We perform evaluations under two settings: by considering (i) word similarity between visual words only and (ii) between all words (column 100% in Table 2). For the models CNN, VAE and their concatenation with SGNS embeddings, the latter setting is not applicable. The two last rows correspond to the multimodal embeddings inferred from our model. In particular, PIXIE$_+$

[2]For instance, Lazaridou et al. (2015) reports only 5,100 visual words, nearly half of what we have defined. Collell et al. (2017) used pre-trained GloVe word vectors obtained from a different corpus.

[3]For each method we use the hyper-parameters recommended by the authors.

(resp. PIXIE$_\oplus$) represents the multimodal embeddings built using Eq. (10) (resp. Eq. (11)).

Overall, we note that PIXIE$_\oplus$ offers the best performance in almost all situations. This provides strong empirical support for the proposed model. Below, we discuss the above results in more depth to better understand them and characterize the circumstances in which our model performs better.

How relevant is our formulation? Except PIXIE and V-SGNS, most of the multimodal competing methods rely on independently pre-computed linguistic embeddings. As Table 2 shows, PIXIE and V-SGNS are often the best performing multimodal models, which provides empirical evidence that accounting for perceptual information while learning word embeddings from text is beneficial. Moreover, the superior performance of PIXIE$_\oplus$ over V-SGNS suggests that our model does a better job at combining perception and language to learn word representations.

Joint learning is beneficial PIXIE$_\oplus$ outperforms VAE⊕SGNS in almost all cases, which demonstrates the importance of joint learning.

Where does our approach perform better? On datasets that focus on *semantic/taxonomic similarity*, our approach dominates all other methods.

On datasets focusing on *general relatedness*, our approach obtains mixed results. While dominating other approaches on MEN, it tends to perform worst than SGNS on MTurk and REL (under the 100% setting). One possible explanation is that general relatedness tends to focus more on "extrapolating" from one word to another word (such as SWAN is related to LAKE), while our approach better models more concrete relationships (such as SWAN is related to GOOSE). The low performance of CNN and VAE confirms this hypothesis.

On the VISSIM dataset focusing on *visual similarity*, both CNN⊕SGNS and VAE⊕SGNS perform the best, strongly suggesting that visual and linguistic data are complementary. Our approach comes very close to these two methods. Note that our learning objective is to jointly explain visual features and word-context co-occurrences. Thus, two visually similar words, which never occur within the same context, could be mapped into slightly different directions in the latent space.

Visual Propagation Here we wish to evaluate the ability of our model to propagate perceptual information to words lacking visual features. To this end, we randomly select a subset of $2,000$ words for which we have visual features, and we train our model under two different settings: the visual features of the selected 2K words (i) are taken into account (PIXIE$_\oplus$), (ii) are ignored, i.e. set to zero (PIXIE$_\oplus{}^{(*)}$). We then perform evaluations, under the two settings, on the datasets of Table 1 considering only pairs composed of words in the above subset of 2K words. As baselines for this experiment, we consider SGNS and the multimodal approaches which can propagate perceptual information, namely V-SGNS and IV-SGNS, as well as their outputs when the 2K visual features are ignored (denoted by V-SGNS$^{(*)}$, and IV-SGNS$^{(*)}$).

Table 3: Spearman's score on subsets of visual words. The symbol $(*)$ indicates the visual features for the subset of 2K words have been ignored when training. Bold highlights the best performing method. Blue color highlights the best performing model under the $(*)$ setting.

	Semantic/taxonomic					Relatedness			Visual	REL +SIM
models	SEMSIM	SimLex	SIM	EN-RG	EN-MC	MEN	REL	MTurk	VISSIM	WORDSIM
SGNS	0.55	0.40	0.67	0.57	0.52	0.68	0.53	0.75	0.41	0.63
V-SGNS	0.61	0.32	**0.72**	0.71	0.69	0.69	0.52	0.75	0.44	**0.66**
IV-SGNS (LINEAR)	0.50	0.40	0.60	0.59	0.60	0.66	0.41	0.72	0.37	0.54
PIXIE$_\oplus$	**0.71**	**0.59**	**0.72**	**0.73**	**0.71**	**0.77**	**0.57**	0.74	**0.60**	0.62
V-SGNS$^{(*)}$	0.58	0.29	**0.72**	0.68	0.61	0.67	0.44	**0.76**	0.43	0.60
IV-SGNS$^{(*)}$	0.50	0.40	0.61	0.58	0.58	0.65	0.40	0.72	0.37	0.54
PIXIE$_\oplus{}^{(*)}$	0.60	0.43	**0.72**	0.65	0.60	0.70	0.56	**0.76**	0.45	0.63

The results are given in Table 3. We observe that PIXIE$_\oplus{}^{(*)}$ outperforms SGNS in almost all cases. Recall that, if we ignore the visual features for all words, PIXIE reduces to SGNS. We can therefore attribute the performance improvement of PIXIE$_\oplus{}^{(*)}$ over SGNS to the propagation of visual information to the subset of 2K words. Compared to multimodal methods, PIXIE$_\oplus{}^{(*)}$ (resp. PIXIE$_\oplus$) performs better than V-SGNS$^{(*)}$ (resp. V-SGNS) and IV-SGNS$^{(*)}$ (resp. IV-SGNS) in almost situations. This suggests that our formulation allows perceptual information to propagate better.

Table 4: Word pair cosine similarity computed based on SGNS and PIXIE$_\oplus{}^{(*)}$ embeddings.

Word pairs	SGNS	PIXIE$_\oplus{}^{(*)}$
(chicken, turkey)	0.35	0.55
(helicopter, jet)	0.63	0.76
(falcon, hawk)	0.49	0.70
(cathedral, chapel)	0.69	0.80
(cup, mug)	0.39	0.46

Table 5: 10 nearest neighboring words to the query words in different embedding spaces generated by different methods. Only "visual" words contain direct visual representations in our dataset. The concreteness score of each query word is reported between parenthesis (see text for details).

	Query word	SGNS	V-SGNS	PIXIE$_{\oplus}$
Visual	goose (4.81)	quail, pig, shark, gull, smoky, sooty, bald, owl, guppy, bird	fowl, quail, duck, bat, bulldog, puppy, warbler, blossom, wolfhound, crows	geese, duck, swans, swan, teal, loon, albatross, ostrich, gull, eider
	brave (1.26)	courageous, young, man, fearless, heroic, thief, horrible, adventures, carefree, cowardly	heroes, valiant, ominous, wondrous, fearless, wanton, sabers, beast, excalibur, carefree	bodyguard, wives, benefactors, womanizer, heroes, valiant, immortal, housewives, fearless, warrior
Non Visual	birthstone (4.25)	heaviest, yeti, koala, snowfall, intrusions, mourning, amalthea, gleaming, incidentally, dolly	emerald, lily, lavender, earthy, olive, delicacy, acacia, belladonna, flower, poppy	beryl, emerald, lily, sandalwood, hula, guinevere, pearls, holly, jasmine, jewels
	savagery (1.73)	swamp, man, crazy, madman, mysterious, thief, jeffrey, rage, mad, hardy	mad, zombies, beast, fabulous, mysterious, nightmare, ghosts, alien, mayhem, bandits	cannibals, evil, legends, werewolves, beast, haunting, ghosts, zombies, mayhem, thrillers

In Table 4, we report the cosine similarity between 5 semantically/visually coherent word pairs (from our subset of 2K words). Although the visual vectors of these words were removed during training, the PIXIE's word embeddings of each pair correlate better as compared to their SGNS counterparts, which provides further support to the propagation of visual information under PIXIE.

4.2.2 Qualitative analysis

Table 5 displays several qualitative examples of word similarity. We have selected 4 words: *goose*, *brave*, *birthstone* and *savagery*. The first two have visual feature representations in our training dataset and the last two do not. Furthermore, for each case, we chose one concrete and one abstract word.[4] For each word, we identify their nearest neighbors in the embedding space.

For the visual words, there is a noticeable difference between our method and others. For instance, for word *goose*, SGNS expresses more "general" relatedness and returns other animals like *pig* or *shark*, while our approach is more specific and tends to give visually similar neighbors by focusing on *goose* looks-like birds. V-SGNS's result is somewhat in between. On the abstract word *brave*, we observe that PIXIE$_{\oplus}$ tends to select more explicit embodiments of the adjective *brave* than SGNS and V-SGNS.

Moving towards the non-visual words, we do not seem to find a consistent discrepancy pattern between V-SGNS and PIXIE$_{\oplus}$, though, as for visual words, both methods seem to select more explicit exemplars compared to SGNS. For instance, for the abstract word *savagery*, both multimodal approaches suggest *cannibals* and *zombies*.

[4]We rely on the concreteness ratings made available by Brysbaert et al. (2014), ranging from 1 to 5.

Table 6: Results for image (I) $\leftrightarrow$ sentence (S) retrieval.

Models	I $\rightarrow$ S			S $\rightarrow$ I		
	K=1	K=5	K=10	K=1	K=5	K=10
SGNS	23.1	49.0	61.6	16.6	41.0	53.8
V-SGNS	21.9	51.7	64.2	16.2	42.0	54.8
IV-SGNS (LINEAR)	22.7	50.5	61.7	17.1	42.6	55.4
PIXIE$_{+}$	24.2	52.5	65.4	17.5	43.8	56.2
PIXIE$_{\oplus}$	**25.7**	**55.7**	**67.7**	**18.4**	**44.9**	**56.9**

4.3 Task 2: Image and Caption Retrieval

We now study the usefulness of the learned word embeddings for the tasks of image and caption retrieval. Our hypothesis is that multimodal word embeddings will perform better for downstream tasks involving multimodal information.

Experimental setup We use the Flickr30K dataset (Young et al., 2014) containing 31,000 images and 155,000 sentences (5 captions per image). The sentences describe the images. The task is to identify the best sentence describing an image or to identify the best image depicting a sentence. We follow the data split setting provided by Karpathy and Fei-Fei (2015), in which 1,000 images are used for validation and 1,000 for testing. The rest is used for training.

The retrieval models compute the proximity between the image features and the sentence embeddings. For image features, we use the precomputed features provided by Faghri et al. (2017), which are extracted from the FC7 layer of VGG-19 (Simonyan and Zisserman, 2014). These 4,096-dimensional features are then linearly mapped to 1,024-dimensional features. For sentences, we use an one GRU-layer over the sequences of the word embeddings, resulting in 1,024-dimensional sentence embeddings.

We use a triplet loss to train the retrieval model such that the inner product between the corresponding image feature and the sentence is greater than the inner products with incorrect sentences (or im-

ages) (Kiros et al., 2014). The linear mapping and the GRU layer are then optimized to minimize the loss. We use the ADAM optimizer with the learning rate of 0.0002 and divide it by 10 every 15 epochs and we train the model for 30 epochs. Note that we do not *fine-tune* either the original visual feature or the word embeddings.

Results Table 6 summarizes the results. The evaluation metrics are accuracies at top-K (K=1, 5, or 10) retrieved sentences or images. Our model consistently outperforms SGNS and other competing multimodal methods, which provides additional support for the benefits of our approach.

5 Conclusion

We propose PIXIE, a novel probabilistic model joining textual and perceptual information to infer multimodal word embeddings. In our model, both linguistic and visual latent factors work in concert to explain the co-occurrences of words and their contexts in a corpus. Empirical results show that our model achieves equally competitive or stronger results when compared to state-of-the-art methods for multimodal embeddings.

Currently our model relies on unsupervised learning to infer visual factors. Explicit knowledge of similar and dissimilar visual categories could potentially disentangle latent factors better for alignment with linguistic data. How to incorporate visual domain knowledge more explicitly into the model would be an interesting direction for future research. While we build on skip-gram, the idea of PIXIE could be extended to other word embedding models, e.g., Glove (Pennington et al., 2014), ELMO (Peters et al., 2018), etc.

Acknowledgements

This work was partially supported by the virtual lab Inria@SiliconValley, ANR Grant GRASP No. ANR-16-CE33-0011-01, and a grant from CPER Nord-Pas de Calais/FEDER DATA Advanced data science and technologies 2015-2020, NSF IIS-1065243, 1451412, 1513966/ 1632803/1833137, 1208500, CCF-1139148, a Google Research Award, an Alfred P. Sloan Research Fellowship, gifts from Facebook and Netflix, and ARO# W911NF-12-1-0241 and W911NF-15-1-0484. We thank anonymous reviewers for their suggestions and comments.

References

Eneko Agirre, Enrique Alfonseca, Keith Hall, Jana Kravalova, Marius Paşca, and Aitor Soroa. 2009. A study on similarity and relatedness using distributional and wordnet-based approaches. In *NAACL HLT*, pages 19–27.

Marco Baroni, Georgiana Dinu, and Germán Kruszewski. 2014. Don't count, predict! a systematic comparison of context-counting vs. context-predicting semantic vectors. In *ACL*, pages 238–247.

Steven Bird, Ewan Klein, and Edward Loper. 2009. *Natural language processing with Python: analyzing text with the natural language toolkit*. O'Reilly Media, Inc.

David M Blei, Alp Kucukelbir, and Jon D McAuliffe. 2017. Variational inference: A review for statisticians. *Journal of the American Statistical Association*, 112(518):859–877.

Elia Bruni, Nam-Khanh Tran, and Marco Baroni. 2014. Multimodal distributional semantics. *Journal of Artificial Intelligence Ressearch*, 49(2014):1–47.

Marc Brysbaert, Amy Beth Warriner, and Victor Kuperman. 2014. Concreteness ratings for 40 thousand generally known english word lemmas. *Behavior research methods*, 46(3):904–911.

Iacer Calixto and Qun Liu. 2017. Incorporating global visual features into attention-based neural machine translation. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, (EMNLP)*, pages 992–1003.

Guillem Collell, Ted Zhang, and Marie-Francine Moens. 2017. Imagined Visual Representations as Multimodal Embeddings. In *AAAI*, pages 4378–4384.

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

Fartash Faghri, David J Fleet, Jamie Ryan Kiros, and Sanja Fidler. 2017. VSE++: Improving Visual-Semantic Embeddings with Hard Negatives. *arXiv preprint arXiv:1707.05612*.

Christiane Fellbaum. 1998. *WordNet*. Wiley Online Library.

Yansong Feng and Mirella Lapata. 2010. Visual information in semantic representation. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 91–99. Association for Computational Linguistics.

Lev Finkelstein, Evgeniy Gabrilovich, Yossi Matias, Ehud Rivlin, Zach Solan, Gadi Wolfman, and Eytan Ruppin. 2001. Placing search in context: The concept revisited. In *WWW*, pages 406–414.

Guy Halawi, Gideon Dror, Evgeniy Gabrilovich, and Yehuda Koren. 2012. Large-scale learning of word relatedness with constraints. In *SIGKDD*, pages 1406–1414.

Zellig S Harris. 1954. Distributional structure. *Word*, 10(2-3):146–162.

Felix Hill, Roi Reichart, and Anna Korhonen. 2015. Simlex-999: Evaluating semantic models with (genuine) similarity estimation. *Computational Linguistics*, 41(4):665–695.

Yangqing Jia, Evan Shelhamer, Jeff Donahue, Sergey Karayev, Jonathan Long, Ross Girshick, Sergio Guadarrama, and Trevor Darrell. 2014. Caffe: Convolutional architecture for fast feature embedding. In *ACM Multimedia*, pages 675–678.

Andrej Karpathy and Li Fei-Fei. 2015. Deep visual-semantic alignments for generating image descriptions. In *CVPR*, pages 3128–3137.

Douwe Kiela and Léon Bottou. 2014. Learning Image Embeddings using Convolutional Neural Networks for Improved Multi-Modal Semantics. In *EMNLP*, pages 36–45.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Diederik P Kingma and Max Welling. 2013. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*.

Ryan Kiros, Ruslan Salakhutdinov, and Richard S Zemel. 2014. Unifying visual-semantic embeddings with multimodal neural language models. *arXiv preprint arXiv:1411.2539*.

Benjamin Klein, Guy Lev, Gil Sadeh, and Lior Wolf. 2015. Associating neural word embeddings with deep image representations using fisher vectors. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4437–4446.

Angeliki Lazaridou, Marco Baroni, et al. 2015. Combining Language and Vision with a Multimodal Skip-gram Model. In *NAACL HLT*, pages 153–163.

Omer Levy and Yoav Goldberg. 2014. Neural word embedding as implicit matrix factorization. In *NIPS*, pages 2177–2185.

Omer Levy, Yoav Goldberg, and Ido Dagan. 2015. Improving distributional similarity with lessons learned from word embeddings. *Transactions of the Association for Computational Linguistics*, 3:211–225.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *NIPS*, pages 3111–3119.

George A Miller and Walter G Charles. 1991. Contextual correlates of semantic similarity. *Language and cognitive processes*, 6(1):1–28.

Nasrin Mostafazadeh, Ishan Misra, Jacob Devlin, Margaret Mitchell, Xiaodong He, and Lucy Vanderwende. 2016. Generating natural questions about an image. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (ACL)*.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. In *EMNLP*, pages 1532–1543.

Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proc. of NAACL*.

Kira Radinsky, Eugene Agichtein, Evgeniy Gabrilovich, and Shaul Markovitch. 2011. A word at a time: computing word relatedness using temporal semantic analysis. In *WWW*, pages 337–346.

Herbert Rubenstein and John B Goodenough. 1965. Contextual correlates of synonymy. *Communications of the ACM*, 8(10):627–633.

Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. 2015. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision*, 115(3):211–252.

Carina Silberer, Vittorio Ferrari, and Mirella Lapata. 2017. Visually grounded meaning representations. *IEEE transactions on pattern analysis and machine intelligence*, 39(11):2284–2297.

Carina Silberer and Mirella Lapata. 2014. Learning Grounded Meaning Representations with Autoencoders. In *ACL*, pages 721–732.

Karen Simonyan and Andrew Zisserman. 2014. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.

Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. 2015. Going deeper with convolutions. In *CVPR*, pages 1–9.

Peter Young, Alice Lai, Micah Hodosh, and Julia Hockenmaier. 2014. From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions. *Transactions of the Association for Computational Linguistics*, 2:67–78.