Learning Zero-Shot Multifaceted Visually Grounded Word Embeddings via Multi-Task Training

Hassan Shahmohammadi, Hendrik P. A. Lensch, and R. Harald Baayen

University of Tübingen
{hassan.shahmohammadi, hendrik.lensch, harald.baayen}@uni-tuebingen.de

Abstract

Language grounding aims at linking the symbolic representation of language (e.g., words) into the rich perceptual knowledge of the outside world. The general approach is to embed both textual and visual information into a common space -the grounded space- confined by an explicit relationship between both modalities. We argue that this approach sacrifices the abstract knowledge obtained from linguistic co-occurrence statistics in the process of acquiring perceptual information. The focus of this paper is to solve this issue by implicitly grounding the word embeddings. Rather than learning two mappings into a joint space, our approach integrates modalities by determining a reversible grounded mapping between the textual and the grounded space by means of multi-task learning. Evaluations on intrinsic and extrinsic tasks show that our embeddings are highly beneficial for both abstract and concrete words. They are strongly correlated with human judgments and outperform previous works on a wide range of benchmarks. Our grounded embeddings are publicly available here.

1 Introduction

The distributional hypothesis asserts that words occurring in similar contexts tend to be similar in meaning (Harris, 1954). Current state-of-the-art word embedding models (Pennington et al., 2014; Peters et al., 2018), despite their successful application to different NLP tasks (Wang et al., 2018), suffer from the limitation of being derived from lexical co-occurrences in written texts without grounding in more general knowledge (Harnad, 1990; Burgess, 2000), such as captured by human perceptual and motor systems (Pulvermüller, 2005; Therriault et al., 2009). To move beyond this limitation, research has been directed to linking word embeddings to perceptual knowledge in visual scenes. Most studies have attempted to bring visual and language representations into close vicinity in a common feature space (Silberer and Lapata, 2014; Kurach et al., 2017; Kiela et al., 2018). However, studies of human cognition indicate that areas of the brain are differentially involved in the processing of abstract and concrete words (Paivio, 1990; Anderson et al., 2017). According to Montefinese (2019), the perirhinal cortex, a region related to memory and recognition, processes both concrete and abstract concepts, whereas the parahippocampal cortex, associated with memory formation, is only responsible for abstract concepts.

We argue that forcing the textual and visual modalities to be represented in a shared space causes grounded embeddings to suffer from the bias towards concrete words reported by Park and Myaeng (2017); Kiela et al. (2018). We therefore propose a novel zero-shot approach that implicitly integrates perceptual knowledge into pre-trained textual embeddings (such as GloVe (Pennington et al., 2014) or fastText (Bojanowski et al., 2017)) via multi-task training. We show that our approach learns multifaceted grounded embeddings which capture multiple aspects of words' meaning and are highly beneficial for both concrete and abstract words.

Figure 1 lays out the architecture of our grounding model. It learns a reversible mapping from pre-trained text-based embeddings to grounded embeddings which maintains the linguistics co-occurrence statistics while augmenting visual information. The architecture features a similar structure as an auto-encoder (Press and Wolf, 2017) translating from words to grounded space and back. The training is carried out as multi-task learning by combining image-induced next word prediction (image-based language model) and image-sentence pair discrimination. At the core is a mapping matrix that acts as an intermediate representation between the grounded and textual space, which learns to visually ground the textual word vectors. This map-

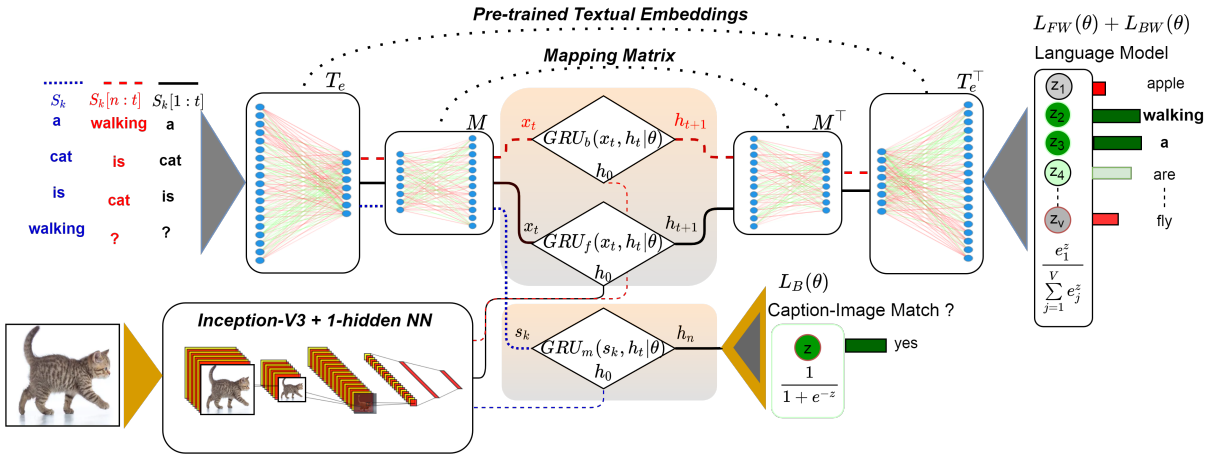

Figure 1: Our zero-shot multi-task learning approach includes: 1. Two GRU based language-model tasks in forward (GRU_f) and backward (GRU_b) directions represented by dashed red and solid black lines in the upper block. 2. A matching task predicting if the given sentence describes the image (blue dotted line, lower block). The zero-shot mapping matrix M shared by all the tasks, learns to visually ground the textual word vectors by learning a reversible mapping from textual space to grounded space.

ping is trained on a subset of words and then is applied to ground the full vocabulary of textual embeddings in a zero-shot manner.

We evaluate our grounded embeddings on both intrinsic and extrinsic tasks (Wang et al., 2019) and show that they outperform textual embeddings and previous related works in the majority of cases. Moreover, by grounding two textual word embedding models, namely GloVe and fastText, we show that our approach generalizes across different textual embedding models. Overall, our contributions are the following:

- We design a language grounding framework that can effectively ground different pre-trained word embeddings.
- Unlike many previous works, our embeddings are not limited to words with concrete manifestation, they support all types of words.
- We provide a model that effectively grounds unseen words in a zero-shot manner.
- We show that visual grounding has the potential to refine the irregularities of text-based vector-space.

2 Related Works

The many attempts to bridge images and text in order to obtain visually grounded word/sentence representations can be grouped into the following categories.

Feature Level Fusion: where the grounded embedding is the result of combining the visual and textual features. Combining strategies range from simple concatenation to adopting SVD and GRU gating mechanism (Bruni et al., 2014; Kiela and Bottou, 2014; Kiros et al., 2018).

Mapping to Perceptual Space: this is usually a regression task predicting the image vector given its corresponding textual vector. The grounded embedding are extracted from an intermediate layer in autoencoders (Silberer and Lapata, 2014; Hasegawa et al., 2017), the output of an MLP (Collell Talleda et al., 2017) or an RNN (Kiela et al., 2018). Another method is mapping both modalities into a common space in which their distance is minimized (Kurach et al., 2017; Park and Myaeng, 2017).

Equipping Distributional Semantic Models with Visual Context: here images are treated as a context in the process of computing the word vectors. Many of this approaches modify the Word2Vec (Mikolov et al., 2013) and GloVe (Pennington et al., 2014) models by incorporating image features to the context for concrete words (Hill and Korhonen, 2014; Kottur et al., 2016; Zablocki et al., 2017; Ailem et al., 2018); minimizing the max-margin loss between the image-vector and its corresponding word vectors (Lazaridou et al., 2015); providing social cues based on child-directed speech along with visual scenes (Lazaridou et al., 2016); or by extracting the relationship between words and images using multi-view spectral graph (Fukui et al., 2017). Another technique is to augment perceptual information using an RNN-based language model (Mao et al., 2016). While this task is similar

to image-captioning, the main idea is that updating the textual word vectors during training will ground them into the associated images.

Hybrid: this category covers the combination of previous methods, and other strategies. Here, the grounded word vectors are usually the results of updating the textual word vectors during training or the output of sentence encoders such as LSTM (Hochreiter and Schmidhuber, 1997). Such methods include predicting the image vector (regression) along with training a language model (Chrupała et al., 2015) or generating an alternative caption at the same time (Kiela et al., 2018). More recently, new approaches such as refining the sentence embeddings based on the relationship of their corresponding images (Bordes et al., 2019) and using the coefficients of classifiers for grounded representation have emerged (Moro et al., 2019). Our model lies in the hybrid category as we take a multitasking approach. However, unlike some previous works (Kiela et al., 2018; Collell Talleda et al., 2017; Bordes et al., 2019) we do not define strict constraints between the image features and their captions. Our model learns the relationship indirectly via multi-task training.

3 Multi-Task Visual Grounding

In this section, we explain the proposed method in details. For $(S_k, I_k) \in D$, assume $S_k = [w_1, w_2...w_n]$ be a sentence with n words describing the image I_k in the dataset D. We use the Microsoft_COCO_2017 dataset (Lin et al., 2014) in our experiments. Let $T_e(w) \in \mathbb{R}^d$ be a pre-trained textual embedding of the word w, which has been trained on textual data only (e.g., GloVe). The objective is to train the mapping matrix $M_{d\times c}$ to ground the word vector $T_e(w)$ to the image I_k and obtain the grounded embedding $G_e(w) \in \mathbb{R}^c$ of the word w. To do so, we train the matrix M to refine the textual vector-space via two image-based language model tasks and a binary discrimination task on image-sentence pairs. For the language models, a GRU (Cho et al., 2014) is trained to predict the next word, given the previous words in a sentence (image caption) and its associated image vector. The output of the textual embedding T_e is used to compute the probability distribution over the vocabulary (see Figure1). We employ an identical scenario to form a second language model task using another GRU, where the sentence is fed backward into the model. The image-sentence discrimination is a binary classification task predicting if the given sentence S_k represented in the grounded space matches the image I_k. This type of discrimination task has been found useful for learning the alignment between modalities (Lu et al., 2019). By training the model simultaneously on these three tasks confined by a linear transformation, We augment the visual information into the grounded embeddings (output of mapping matrix in Figure 1) while preserving the underlying structure of the textual embeddings.

3.1 Language Model

Given the input caption associated with image I_k as $S_k = [w_1, w_2...w_n]$, we first encode the words using a pre-trained textual embedding T_e to obtain the embeddings as $S_t = [t_1, t_2...t_n]$. We then linearly project these embeddings from textual space into the visually grounded space via the trainable mapping matrix M as follows:

$$G_e(S_k) = S_{t_{n\times d}} \times M_{d\times c} \quad (1)$$

to obtain a series of grounded vectors $G_e(S_k) = [x_1, x_2...x_n]$ where $x_i \in \mathbb{R}^c$. In the grounded space, the perceptual information of the image I_k corresponding to S_k is fused using a single-layer GRU that predicts the next output $h_{t+1} = GRU_f(x_t, h_t|\theta)$, where θ denotes the trainable parameters, x_t the current input ($G_e(w_t)$), and $h_t \in \mathbb{R}^c$ the current hidden state.

Image information is included by initializing the first hidden state h_0 by the image vector of I_k. That is, for each image, we extract the $2048d$ feature vector of the penultimate layer of Inception-V3 (Szegedy et al., 2016) trained on ImageNet (Deng et al., 2009). The image features are then mapped into the hidden state h_0 by using a one-hidden network layer with $tanh$ activation. The GRU update gate enables the model to decide how to propagate perceptual knowledge into the mapping matrix. In fact, this has been shown to be more effective than providing the image vector at each time step as input (Mao et al., 2016).

The transpose of the mapping matrix ($M^\top$) is used to map back from grounded space to the textual space. That is, the output of the GRU in each time-step is mapped back into the textual space as:

$$w_{next} = h_t \times M^\top, \quad (2)$$

where $w_{next} \in \mathbb{R}^d$ is an approximation of the next word's textual embedding. The mapping matrix

M is used to both encodes and decodes into/from the grounded space. This improves generalization (Press and Wolf, 2017) and prevents vanishing gradient problem compared to the case where the mapping matrix is only used at the beginning of the network (Mao et al., 2016). w_{next} is fed into the transpose of the textual embeddings in the same scenario: $z = T_e^\top(w_{next})$, where $z \in \mathbb{R}^V$ is the vector with the size of the vocabulary. The final probability distribution is computed by a softmax:

$$P(y = j|z) = \frac{e^{z^\top W_j}}{\sum_{k=1}^{V} e^{z^\top W_k}} \quad (3)$$

Defining the input (previous words and the image vector) and the predicted output (next word prediction) as above, we minimize the categorical cross entropy which is computed for each batch as:

$$\mathcal{L}_{FW}(\theta) = -\frac{1}{B}\sum_{i=1}^{B}\sum_{c=1}^{V} y_{i,c} log(\hat{y}_{i,c}), \quad (4)$$

where V is the size of the vocabulary and B indicates the batch-size. $\hat{y}_{i,c}$ and $y_{i,c}$ are the predicted probability and ground truth for sample i with respect to the class c.

Moreover, we define a second similar task: Given the input caption associated with image I_k as $S_k = [w_1, w_2 ... w_n]$, we reverse the order of the words: $S_k = [w_n, w_{n-1} ... w_1]$ and use another GRU (GRU_b in Figure 1) with identical structure trained on the loss $\mathcal{L}_{BW}(\theta)$. The rest of the network is shared between these two tasks. Having this backward language-model is analogous to bi-directional (Schuster and Paliwal, 1997) GRUs which, however, can not be used directly since the ground truth would be exposed by operating in both directions.

3.2 Image-sentence discrimination

Even though context-driven word representations are a powerful way to obtain word embeddings (Pennington et al., 2014; Peters et al., 2018), the performance of such models varies largely on language-vison tasks such as image-caption retrieval (Burns et al., 2019). Therefore, we propose yet another task to assure that the grounded embeddings learn the association between words and images. A discrimination task predicts if the given image and sentence describe the same content or not (shown by 'caption-image match?' in Figure 1).

Given the input caption for image I_k as $S_k = [w_1, w_2 ... w_n]$, after projecting the embeddings into the grounded space as before, we encode the whole sentence by employing a third single-layer GRU (GRU_m in Figure 1) with the same structure as before $h_n = GRU_m(G_e(S_k), h_0|\theta)$. However, this time only the last output h_n encoding the whole sentence is considered. h_0 is again initialized with the image vector of I_k. The final output is computed by a sigmoid function. This task shares the matrix M, textual embeddings T_e, and the one hidden neural layer for mapping the image vectors into the hidden state. We minimize the binary cross entropy, which could be computed for each batch as:

$$\mathcal{L}_B(\theta) = -\frac{1}{B}\sum_{i=1}^{B} y_i log(\hat{y}_i) + (1 - y_i) log(1 - \hat{y}_i), \quad (5)$$

where $\hat{y}$ and y refer to the predicted probability and ground truth respectively. For negative mining, half of the captions in each batch are replaced with captions of different, random images.

3.3 Regularization and overall loss

All the three tasks explained above share the pre-trained textual embeddings (see Figure 1) which gives rise to the question if the textual embeddings should be updated or kept fixed during training. By updating, they might deviate from the original vectors and disturb the pre-trained semantic relations, especially given our limited training data. Keeping them fixed, on the other hand, does not provide the flexibility to generate the desired grounding as these embeddings are noisy and not perfect (Yu et al., 2017). To prevent disturbing the semantic information of words and having the flexibility, we propose the following regularization on the embedding matrix:

$$\mathcal{R}(\alpha, \beta) = \frac{\alpha}{|V|}\sum_{w \in V}\left|\beta - \frac{w_f \cdot w_u}{\|w_f\| \times \|w_u\|}\right|, \quad (6)$$

where α controls the overall impact and β controls how much the new word vectors w_u are allowed to deviate from the pre-trained embedding w_f.

We join all the tasks to a single model and minimize the following loss:

$$\mathcal{L}_{All}(\Theta) = \mathcal{L}_{FW}(\theta) + \mathcal{L}_{BW}(\theta) + \mathcal{L}_B(\theta) + \mathcal{R}(\alpha, \beta) \quad (7)$$

where Θ is all the trainable parameters.

4 Experimental setup

We use the Microsoft_COCO_2017 dataset (Lin et al., 2014) for training. Each sample contains an image with 5 captions. The whole dataset is split into $118k$ train and $5k$ validation samples. Each batch includes 256 image vectors along with one of their captions. Hence, multiple image vectors might occur in each batch. Image vectors are obtained by transferring the penultimate layer of pre-trained Inception-V3 (Szegedy et al., 2016) trained on ImageNet (Deng et al., 2009). One-hidden NN with $tanh$ activation function is employed to project the image vectors into the initial hidden state of the GRUs: $h_t \in \mathbb{R}^{1024}$. We lowercase all the words, delete the punctuation marks, and only keep the top $10k$ most frequent words. Two popular pre-trained textual word embeddings namely GloVe ($crawl-300d-2.2M$) and fastText ($crawl-300d-2M$) are used for initialization of the embedding T_e. The mapping matrix M transforms the textual embeddings into the grounded space. We investigate the best dimension of this step and the improvement over pure textual embeddings in the next sections. Batch normalization (Ioffe and Szegedy, 2015) is applied after each GRU. For the regularization, $\mathcal{R}(\alpha = 0.001, \beta = 1)$ for GloVe and $\mathcal{R}(\alpha = 0.01, \beta = 0)$ for fastText yielded the best relative results by meta parameter search. We trained the model for 20 epochs with 5 epochs tolerance early stopping using NAdam (Dozat, 2016) with a learning rate of 0.001.

As we train a single mapping matrix M for projecting from textual to grounded space, it can be used after the training to transfer out-of-vocabulary (OoV) word-vectors into the grounded space as well. This zero-shot transformation is then applied to obtain the visually grounded version of the Glove and fastText despite being exposed to only $10k$ words.

5 Evaluations

Despite the advance of current word embedding models and ample NLP tasks, the question of what is a good embedding model remains open (Wang et al., 2019). There are two main categories of evaluation methods: intrinsic and extrinsic. Intrinsic evaluators measure the quality of word embeddings independent of any downstream tasks. One common task of such evaluators is word similarity which aims at scoring a pair of words based on their semantic equivalence. Extrinsic evaluators on the other hand assess the performance based on sentence-level downstream tasks. Moreover, one might argue that there is not necessarily a positive correlation between intrinsic and extrinsic methods for a single word embedding model (Wang et al., 2019). We use both types of evaluators and compare the results of our visually grounded embeddings to their counterpart textual embeddings and previous related works. To do so, we first create a visually grounded version of both GloVe and fastText using our zero-shot mapping matrix M (see Section 4) and run the following evaluation methods:

Intrinsic Evaluators: we evaluate on some of the common lexical semantic similarity benchmarks namely MEN (Bruni et al., 2014), SimLex999 (Hill et al., 2015), Rare-Words (Luong et al., 2013), MTurk771 (Halawi et al., 2012), WordSim353 (Finkelstein et al., 2001), and SimVerb3500 (Gerz et al., 2016). The evaluation metric is the Spearman correlation between the predicted cosine similarity vector and the ground truth.

Extrinsic Evaluators: We evaluate on the semantic similarity benchmarks from year 2012 to 2016 using SentEval (Conneau and Kiela, 2018). Here, the task is to measure the semantic equivalence of a pair of sentences solely based on their cosine coefficient. The evaluation metric is the spearman correlation between the predicted and the ground truth similarity vector. We are particularly interested in these benchmarks because they do not provide any training data. Thus, they evaluate the generalization power of the given vector-space. We averaged all the word-vectors in each sentence to obtain the sentence representation. While averaging is a simple sentence encoder, it is a great tool to evaluate the underlying structure of a vector-space and find its irregularities. For instance, the representation of a pair of sentences such as 'her dog is very smart' and 'his cat is too dumb' are very similar in a vector space in which dissimilar but related words (e.g., smart and dumb) are clustered together. We will show that our model refines the textual vector-space and alleviate such irregularities.

6 Results

Intrinsic Evaluation: The upper part of table 1 shows the intrinsic evaluation results for the pre-trained GloVe, fastText, and their visually grounded versions. In general, fastText performs

Model	RW	MEN	WSim 353	MTurk 771	SimVerb 3500	SimLex 999	Mean
GloVe	45.5	80.5	73.8	71.5	28.3	40.8	56.7
V_GloVe	52.6	**85.1**	**78.9**	73.4	37.4	51.8	**63.2**
fastText	56.1	81.5	72.2	75.1	37.8	47.1	61.6
V_fastText	**57.8**	83.6	73.9	**76.1**	**39.2**	49.0	**63.2**
Cap2Both	48.7	81.9	71.2	_	_	46.7	_
Cap2Img	52.3	84.5	75.3	_	_	51.5	_
Park et al.	_	83.8	77.5	_	_	**58.0**	_
Collell et al.	_	81.3	_	_	28.6	41.0	_

Table 1: Intrinsic evaluation. Visual grounding (denoted by 'V') improves results compared to pre-trained fastText and GloVe on all test sets.

better on word-level tasks compared to GloVe, probably because it provides more context for each word by leveraging from its sub-words.
By the proposed visual grounding, significant improvements are achieved on *all* datasets for both fastText and GloVe, even on pure lexical tasks. Analyzing why the improvement varies across different datasets is difficult. However, the table reveals interesting properties. For instance, the improvement on SimLex999 which focuses more on the similarity between words is larger than that on WSim353 with the focus on relatedness. Hence, visual grounding seems to contribute more to similarity than relatedness. Considering the overall performance, visual grounding enhances both embeddings to the same level despite their fundamental differences.

We compare our model to related grounded embeddings by (Collell Talleda et al., 2017; Park and Myaeng, 2017; Kiros et al., 2018; Kiela et al., 2018) (Table 1, bottom). We limit our comparison to those who adopted the pre-trained GloVe or fastText since these pre-trained models alone outperform many visually grounded embeddings such as (Hasegawa et al., 2017; Zablocki et al., 2017) on most of our evaluation datasets.

Conceptually, Kiela et al. (2018) also induces visual grounding on GloVe by using the MSCOCO data set. Even though they propose a number of tasks for training (Cap2Img: predicting the image vector from its caption, Cap2Cap: Generate an alternative caption of the same image, given one of its associated captions; Cap2Both: training by Cap2Cap and Cap2Img simultaneously) our model clearly outperforms them.

Park and Myaeng (2017) proposed a polymodal approach by creating and combining six different types of embeddings for each word. Even though they used two pre-trained embeddings (GloVe and Word2vec) and other resources, our model still outperforms their approach on MEN and WSim353, but their approach is better on Simlex999. Overall, our approach benefits from capturing different perspectives of the words' meaning by learning the reversible mapping in the context of multi-task learning.

Fine-Grained Intrinsic Evaluation: To further investigate the contribution of grounded embeddings, we evaluate our model on the different categories of SimLex999. This dataset is divided into nine sections: All (the whole dataset), adjectives, nouns, verbs, concreteness quartiles (from 1 to 4 increasing the degree of concreteness), and hard pairs. The hard section indicates 333 pairs whose similarity is hard to discriminate from relatedness. The results for our best embeddings on SimLex999 (V_GloVe) is shown in Table 2. We see a large improvement over GloVe in all categories. Some previous works such as (Park and Myaeng, 2017) concluded that perceptual information is beneficial only to concrete words (e.g., apple, table) and adversely effects the abstract words (e.g., happy, freedom). Nevertheless, our model manages to keep the co-occurrence statistics (from the textual model) while augmenting perceptual information, which generalizes to abstract words as well. Therefore, it outperforms GloVe not only on concrete pairs (conc-q4) but also on highly abstract pairs (conc-q1).
We compared the results on SimLex999 with another recent visually grounded model called Picturebook (Kiros et al., 2018) which employs a multi-modal gating mechanism (similar to LSTM and GRU update gate) to fuse the Glove and Picturebook embeddings (Table 2). It uses image feature vectors pre-trained on a fine-grained similarity task with 100+ million images (Wang et al., 2014). Picturebook's performance is highly biased toward concrete words (conc-q3, conc-q4) and performs worse than GloVe by nearly 29% on highly abstract words (conc-q1). Picturebook + GloVe on the other hand shows better results but still performs worse on highly abstract words and adjectives. Our model (V_GloVe) however, can generalize across different categories and outperforms Picturebook+Glove with a large margin on most of the categories while being quite comparable on the others.

Extrinsic Evaluation: Table 3 shows the results

Model	All	Adjs	Nouns	Verbs	Conc-q1	Conc-q2	Conc-q3	Conc-q4	Hard
GloVe	40.8	62.2	42.8	19.6	43.3	41.6	42.3	40.2	27.2
V_GloVe	**51.8**	**72.1**	52.0	**35**	**53.1**	**54.8**	47.4	56.8	**38.3**
Picturebook	37.3	11.7	48.2	17.3	14.4	27.5	46.2	60.7	28.8
Picturebook+GloVe	45.5	46.2	**52.1**	22.8	36.7	41.7	**50.4**	**57.3**	32.5

Table 2: SimLex999 (Spearman's ρ) results. The best result in each category is bolded. Conc-q1 and Conc-q4 contain the most abstract and concrete words respectively. Our embeddings (V_GloVe) generalize across different word types and strongly outperform all the others on most of the categories.

Model	STS12	STS13	STS14	STS15	STS16	Mean
GloVe	53.22	54.14	55.41	60.08	51.43	54.85
V_Glove	**56.17**	**62.42**	**66.57**	**69.91**	**65.56**	**64.13**
Fasttext	21.02	30.36	29.54	39.21	30.10	30.05
V_Fasttext	**31.00**	**38.63**	**42.12**	**51.74**	**38.71**	**40.44**

Table 3: Spearman correlation results for semantic similarly benchmarks. Both grounded embeddings strongly outperform their textual versions on all the tasks.

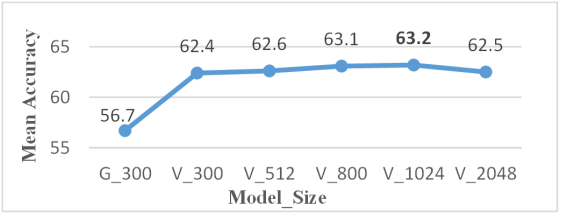

Figure 2: Effect of grounded word-vectors magnitude on intrinsic evaluation. 'G' and 'V' refers to Glove and Visual_Glove respectively. Significant improvement is achieved even with the same size as the lexical GloVe embeddings.

on semantic similarity benchmarks. Both grounded embeddings strongly outperform their textual version on *all* the benchmarks. While fastText outperforms GloVe on intrinsic tasks, GloVe is superior here. The reason might be that fastText dives deeper into words by considering n-grams while GloVe treats each word as a single unit. This might cause fastText to be oblivious of high-level structure of words. Considering the mean score, our model boosts both embeddings approximately by 10 percent.

7 Analysis

We further analyze the performance of our model from different perspectives as follows.

Dependency on the Encoding Dimension c: We train our model with different dimensions of the grounded embeddings and measure the mean accuracy of all the intrinsic datasets. Figure 2 shows the results using GloVe and V_GloVe with different sizes. Significant improvement is already achieved keeping the original dimension of GloVe (300). Higher dimensions up to a certain threshold (1024) increase the accuracy but beyond this point, the model starts to overfit.

Dependency on the Textual Embeddings: In this experiment, we analyze how much of GloVe's original properties are maintained by the visual grounding. Given V_e and G_e as the V_GloVe and GloVe vectors for the word w, we first create a vector containing both embeddings $E_w = [(1-\alpha)G_e; \alpha V_e]$. Varying the relative weight $\alpha \in (0, 1]$ we evaluate on the intrinsic datasets in Table 5. Three of the datasets yield the best results using only the grounded embeddings. The reduction in accuracy regarding 'MEN' is also very subtle. On 'WSim353' and 'Mturk771', however, the best results are achieved with $\alpha \approx 0.5$. This might be because these datasets focus on the relatedness of words compared to SimLex999 for instance, which clearly distinguishes between similarity and relatedness.

Refining the Textual Vector-space: Our created grounded embeddings while improving the relatedness score, priorities similarity over relatedness. This is further demonstrated when looking up nearest neighbors (Table 4). Given the word 'bird', GloVe returns 'turtle' and 'nest' while ours returns 'sparrow' and 'avian' which both reference birds. Moreover, our embeddings retrieve more meaningful words regardless of the degree of abstractness. For the word 'happy' for example, GloVe suffers from a bias toward dissimilar words with high co-occurrence such as 'everyone', 'always', and 'wish'. This issue lays in the fundamental assumption of the distributional hypothesis that words in the same context tend to be semantically related. Therefore, Glove embeddings see antonym words such as 'smart' and 'dump' very similar despite being trained on 840 billion tokens. In addition, common misspelling cases of words (e.g., together) while serving the same role, occur with different frequencies. Hence, they are pulled apart in text-based vector-space. However,

happy		sad		big		bird		horse		together		smart	
G	V	G	V	G	V	G	V	G	V	G	V	G	V
lucky	pleased	sadly	saddened	hard	humongous	turtle	sparrow	dog	racehorse	well	togeather	sensible	witty
everyone	delighted	shame	tragic	little	Big	nest	Birds	riding	Thoroughbred	bring	togheter	dumb	shrewd
love	merry	horrible	mournful			squirrel	avian	ponies	Horses	both	toegther	sophisticated	inteligent
always	thrilled	scared	saddening					donkey	steed	they	togather	attractive	resourceful
wish	joyful	awful	sorrowful							apart	togethor	wise	quick-witted
hope	hapy	pity	Sad							up	2gether		
		kinda	heartbreaking							them	togehter		
		sorry	heartbroken							put	togther		
										along	toghether		
										with	gether		

Table 4: Results of 10 nearest neighbors for Glove (G) and V_Glove (V). Only the differing neighbors are reported. While GloVe retrieves more related words, ours(V_Glove) focuses on similar words. Overall, V_Glove is closer to human judgment and retrieves highly semantically similar words.

Dataset	**Best** α	**Acc. with best** α	**Acc. with** $\alpha = 1$
RareWords	1.00	52.6	52.6
MEN	0.63	85.2	85.1
WSim353	0.57	79.3	78.9
Mturk771	0.52	74.2	73.4
SimVerb3500	1.00	37.4	37.4
SimLex999	1.00	51.8	51.8

Table 5: Sensitivity analysis (Spearman's ρ) on intrinsic datasets. $\alpha = 1$ indicates no use of GloVe and $\alpha = 0$ means no use of V_GloVe. V_Glove alone yield the best results on 3 of the datasets.

Embeddings	$\mathcal{L}_{\mathbf{FW}}$	$\mathcal{L}_{\mathbf{FW}} + \mathcal{L}_{\mathbf{BW}}$	$\mathcal{L}_{\mathbf{FW}} + \mathcal{L}_{\mathbf{BW}} + \mathcal{L}_{\mathbf{B}}$	$\mathcal{L}_{\mathbf{All}}$
V_Glove	61.60	61.82	62.66	63.20
V_fastText	61.70	61.83	61.60	63.20

Table 6: Mean score (Spearman's ρ) on intrinsic datasets with repsect to each task. $\mathcal{L}_{All}$ inidcates all the tasks with regularization loss.

our visual grounding model clearly put them in the same cluster. Our model therefore seems to refine the text-based vector-space and align it with real-world relations. This refinement generalizes to all the words by using our zero-shot mapping matrix which explains the improvement on highly abstract words.

Ablation Study: We further analyse the contribution of each task by performing an ablation evaluation. Table 6 shows the mean score on all the intrinsic datasets (see Table 1) with respect to each loss for both embeddings. While both GloVe and FastText show the same behaviour for language model tasks, fastText embeddings require more deviation ($\beta = 0$ in $\mathcal{R}(\alpha, \beta)$) to adapt to the binary discrimination task ($\mathcal{L}_{BW}$). Textual embeddings T_e were frozen for all the cases except for $\mathcal{L}_{All}$.

8 Conclusion

We investigated the effect of integrating perceptual knowledge from images into word embedding models via multi-task training. We constructed the visually grounded versions of GloVe and fastText by learning a zero-shot transformation from textual to grounded space trained on the MSCOCO dataset. Results on intrinsic and extrinsic evaluation support that images provide a perceptual context that benefits current textual embeddings. The major findings in our experiments are as follows: a) The improvement of visual grounding is not limited to words with concrete meanings but also covers highly abstract words as well. b) The discrimination of relatedness and similarity is more obvious in grounded embeddings. c) Perceptual knowledge can be transferred to purely textual downstream tasks.

Moreover, by analyzing the nearest neighbors of words, we showed that visual grounding is able to refine the irregularities in textual vector-space by aligning the words with their real-world relations. This paves the way for future research on how visual grounding could resolve the problem of dissimilar words that occur frequently in the same context (e.g., small and big). In the future, we will train our model on larger datasets and investigate whether transformer blocks could profitably replace the GRU cells since they lead the state-of-the-art in many downstream tasks. Moreover, while thus far our focus has been on grounding word embeddings, a similar approach could be extended to obtain grounded sentence representations.

Acknowledgements

This project was funded by EXC number 2064/1 – Project number 390727645. The authors thank the International Max Planck Research School for Intelligent Systems (IMPRS-IS) for supporting Hassan Shahmohammadi.

References

Melissa Ailem, Bowen Zhang, Aurelien Bellet, Pascal Denis, and Fei Sha. 2018. A probabilistic model for joint learning of word embeddings from texts and images. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1478–1487, Brussels, Belgium. Association for Computational Linguistics.

Andrew J. Anderson, Douwe Kiela, Stephen Clark, and Massimo Poesio. 2017. Visually grounded and textual semantic models differentially decode brain activity associated with concrete and abstract nouns. *Transactions of the Association for Computational Linguistics*, 5:17–30.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146.

Patrick Bordes, Eloi Zablocki, Laure Soulier, Benjamin Piwowarski, and Patrick Gallinari. 2019. Incorporating visual semantics into sentence representations within a grounded space. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 696–707, Hong Kong, China. Association for Computational Linguistics.

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

Curt Burgess. 2000. Theory and operational definitions in computational memory models: A response to glenberg and robertson. *Journal of Memory and Language*, 43(3):402–408.

Andrea Burns, Reuben Tan, Kate Saenko, Stan Sclaroff, and Bryan A Plummer. 2019. Language features matter: Effective language representations for vision-language tasks. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 7474–7483.

Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using rnn encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*.

Grzegorz Chrupała, Ákos Kádár, and Afra Alishahi. 2015. Learning language through pictures. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 112–118, Beijing, China. Association for Computational Linguistics.

Guillem Collell Talleda, Teddy Zhang, and Marie-Francine Moens. 2017. Imagined visual representations as multimodal embeddings. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence (AAAI-17)*, pages 4378–4384. AAAI.

Alexis Conneau and Douwe Kiela. 2018. SentEval: An evaluation toolkit for universal sentence representations. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. 2009. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee.

Timothy Dozat. 2016. Incorporating nesterov momentum into adam.

Lev Finkelstein, Evgeniy Gabrilovich, Yossi Matias, Ehud Rivlin, Zach Solan, Gadi Wolfman, and Eytan Ruppin. 2001. Placing search in context: The concept revisited. In *Proceedings of the 10th international conference on World Wide Web*, pages 406–414.

Kazuki Fukui, Takamasa Oshikiri, and Hidetoshi Shimodaira. 2017. Spectral graph-based method of multimodal word embedding. In *Proceedings of TextGraphs-11: the Workshop on Graph-based Methods for Natural Language Processing*, pages 39–44, Vancouver, Canada. Association for Computational Linguistics.

Daniela Gerz, Ivan Vulić, Felix Hill, Roi Reichart, and Anna Korhonen. 2016. SimVerb-3500: A large-scale evaluation set of verb similarity. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2173–2182, Austin, Texas. Association for Computational Linguistics.

Guy Halawi, Gideon Dror, Evgeniy Gabrilovich, and Yehuda Koren. 2012. Large-scale learning of word relatedness with constraints. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1406–1414.

Stevan Harnad. 1990. The symbol grounding problem. *Physica D: Nonlinear Phenomena*, 42(1-3):335–346.

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

Mika Hasegawa, Tetsunori Kobayashi, and Yoshihiko Hayashi. 2017. Incorporating visual features into word embeddings: A bimodal autoencoder-based approach. In *IWCS 2017 — 12th International Conference on Computational Semantics — Short papers*.

Felix Hill and Anna Korhonen. 2014. Learning abstract concept embeddings from multi-modal data: Since you probably can't see what I mean. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 255–265, Doha, Qatar. Association for Computational Linguistics.

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

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9(8):1735–1780.

Sergey Ioffe and Christian Szegedy. 2015. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*.

Douwe Kiela and Léon Bottou. 2014. Learning image embeddings using convolutional neural networks for improved multi-modal semantics. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 36–45, Doha, Qatar. Association for Computational Linguistics.

Douwe Kiela, Alexis Conneau, Allan Jabri, and Maximilian Nickel. 2018. Learning visually grounded sentence 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)*, pages 408–418, New Orleans, Louisiana. Association for Computational Linguistics.

Jamie Kiros, William Chan, and Geoffrey Hinton. 2018. Illustrative language understanding: Large-scale visual grounding with image search. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 922–933, Melbourne, Australia. Association for Computational Linguistics.

Satwik Kottur, Ramakrishna Vedantam, José MF Moura, and Devi Parikh. 2016. Visual word2vec (vis-w2v): Learning visually grounded word embeddings using abstract scenes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4985–4994.

Karol Kurach, Sylvain Gelly, Michal Jastrzebski, Philip Haeusser, Olivier Teytaud, Damien Vincent, and Olivier Bousquet. 2017. Better text understanding through image-to-text transfer. *arXiv preprint arXiv:1705.08386*.

Angeliki Lazaridou, Grzegorz Chrupała, Raquel Fernández, and Marco Baroni. 2016. Multimodal semantic learning from child-directed input. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 387–392, San Diego, California. Association for Computational Linguistics.

Angeliki Lazaridou, Nghia The Pham, and Marco Baroni. 2015. Combining language and vision with a multimodal skip-gram model. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 153–163, Denver, Colorado. Association for Computational Linguistics.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. 2014. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer.

Jiasen Lu, Dhruv Batra, Devi Parikh, and Stefan Lee. 2019. Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. *arXiv preprint arXiv:1908.02265*.

Thang Luong, Richard Socher, and Christopher Manning. 2013. Better word representations with recursive neural networks for morphology. In *Proceedings of the Seventeenth Conference on Computational Natural Language Learning*, pages 104–113, Sofia, Bulgaria. Association for Computational Linguistics.

Junhua Mao, Jiajing Xu, Kevin Jing, and Alan L Yuille. 2016. Training and evaluating multimodal word embeddings with large-scale web annotated images. In *Advances in neural information processing systems*, pages 442–450.

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

Maria Montefinese. 2019. Semantic representation of abstract and concrete words: a minireview of neural evidence. *Journal of neurophysiology*, 121(5):1585–1587.

Daniele Moro, Stacy Black, and Casey Kennington. 2019. Composing and embedding the words-as-classifiers model of grounded semantics. *arXiv preprint arXiv:1911.03283*.

Allan Paivio. 1990. *Mental representations: A dual coding approach*. Oxford University Press.

Joohee Park and Sung-hyon Myaeng. 2017. A computational study on word meanings and their distributed representations via polymodal embedding. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 214–223, Taipei, Taiwan. Asian Federation of Natural Language Processing.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. GloVe: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, Doha, Qatar. Association for Computational Linguistics.

Matthew Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. 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)*, pages 2227–2237, New Orleans, Louisiana. Association for Computational Linguistics.

Ofir Press and Lior Wolf. 2017. Using the output embedding to improve language models. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 157–163, Valencia, Spain. Association for Computational Linguistics.

Friedemann Pulvermüller. 2005. Brain mechanisms linking language and action. *Nature reviews neuroscience*, 6(7):576–582.

Mike Schuster and Kuldip K Paliwal. 1997. Bidirectional recurrent neural networks. *IEEE transactions on Signal Processing*, 45(11):2673–2681.

Carina Silberer and Mirella Lapata. 2014. Learning grounded meaning representations with autoencoders. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 721–732, Baltimore, Maryland. Association for Computational Linguistics.

Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. 2016. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2818–2826.

David J Therriault, Richard H Yaxley, and Rolf A Zwaan. 2009. The role of color diagnosticity in object recognition and representation. *Cognitive Processing*, 10(4):335.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2018. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 353–355, Brussels, Belgium. Association for Computational Linguistics.

Bin Wang, Angela Wang, Fenxiao Chen, Yuncheng Wang, and C-C Jay Kuo. 2019. Evaluating word embedding models: Methods and experimental results. *APSIPA transactions on signal and information processing*, 8.

Jiang Wang, Yang Song, Thomas Leung, Chuck Rosenberg, Jingbin Wang, James Philbin, Bo Chen, and Ying Wu. 2014. Learning fine-grained image similarity with deep ranking. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1386–1393.

Liang-Chih Yu, Jin Wang, K. Robert Lai, and Xuejie Zhang. 2017. Refining word embeddings for sentiment analysis. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 534–539, Copenhagen, Denmark. Association for Computational Linguistics.

Eloi Zablocki, Benjamin Piwowarski, Laure Soulier, and Patrick Gallinari. 2017. Learning multi-modal word representation grounded in visual context. *arXiv preprint arXiv:1711.03483*.