Neural Sequential Phrase Grounding (SeqGROUND)

Pelin Dogan[1] Leonid Sigal[2,3] Markus Gross[1,4]
[1]ETH Zürich [2]University of British Columbia [3]Vector Institute [4]Disney Research
{pelin.dogan, grossm}@inf.ethz.ch, lsigal@cs.ubc.ca

Abstract

We propose an end-to-end approach for phrase grounding in images. Unlike prior methods that typically attempt to ground each phrase independently by building an image-text embedding, our architecture formulates grounding of multiple phrases as a sequential and contextual process. Specifically, we encode region proposals and all phrases into two stacks of LSTM cells, along with so-far grounded phrase-region pairs. These LSTM stacks collectively capture context for grounding of the next phrase. The resulting architecture, which we call SeqGROUND, supports many-to-many matching by allowing an image region to be matched to multiple phrases and vice versa. We show competitive performance on the Flickr30K benchmark dataset and, through ablation studies, validate the efficacy of sequential grounding as well as individual design choices in our model architecture.

1. Introduction

In recent years, computer vision has made significant progress in standard recognition tasks, such as image classification [24], object detection [35, 36], and segmentation [4]; as well as in more expressive tasks that combine language and vision. Phrase grounding [33, 48, 49, 58], a task of localizing a given natural language phrase in an image, has recently gained research attention. This constituent task, that generalizes object detection/segmentation, has a breadth of applications that span image captioning [17, 18, 52], image retrieval [12], visual question answering [1, 10, 42], and referential expression generation [16, 21, 26, 27].

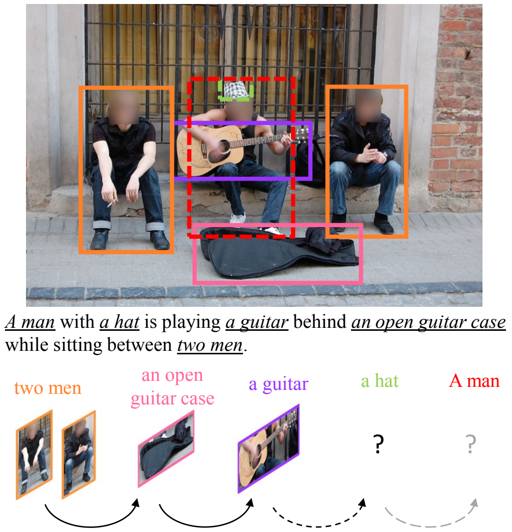

A man with *a hat* is playing *a guitar* behind *an open guitar case* while sitting between *two men*.
two men
an open guitar case
a guitar
a hat
A man
?
?

Figure 1: **Illustration of SeqGROUND.** The proposed neural architecture performs phrase grounding sequentially. It uses the previously grounded phrase-image content to inform the next grounding decision (in reverse *lexical* order).

While significant progress has been made in phrase grounding, stemming from release of several benchmark datasets [21, 23, 27, 34] and various neural algorithmic designs, the problem is far from being solved. Most, if not all, existing phrase grounding models can be categorized into two classes: attention-based [49] or region-embedding-based [32, 58]. In the former, neural attention mechanisms are used to localize the phrases by, typically, predicting a course-resolution mask (*e.g.*, over the last convolutional layer of VGG [39] or another CNN network [14]). In the latter, the traditional object detection paradigm is followed by first detecting proposal regions and then measuring a (typically learned) similarity of each of these regions to the given language phrase. Importantly, both of these classes of models consider grounding of individual phrases individually (or independently), lacking the ability to take into account visual and, often, lingual context and/or reasoning that may exist among multiple constituent phrases.

Consider image grounding noun phrases from a given sentence: "*A lady sitting on a colorful decoration with a bouquet of flowers, that match her hair, in her hand.*" Note that while multiple *ladies* may be present in the image, the grounding of "*a colorful decoration*" uniquely disambiguates to which of these instances the phrase "*A lady*" should be grounded to. While contextual reference in the above example is spatial, other context, including visual maybe useful, *e.g.*, between "*her hair*" and "*a bouquet of flowers*".

Conceptually similar contextual relations exist in object detection and have just started to be explored through the use of spatial memory [5] and convolutional graph networks (CGNNs) [6, 54]. Most assume orderless graph relationships among objects with transitive reasoning. In phrase grounding, on the other hand, the sentence, from which phrases are extracted, may provide implicate linguistic space- and time-order [13]. We show that such ordering is useful as a proxy for sequentially contextualizing phrase grounding decisions. In other words, the phrase that appears *last* in the sentence is grounded first and is used as context for the next phrase grounding in *reverse* lexical order. This explicitly sequential process is illustrated in Figure 1. To our knowledge, our paper is the first to explore such sequential mechanism and architecture for phrase grounding.

Expanding on the class of recent temporal alignment networks (*e.g*., NeuMATCH [7]), that propose neural architectures where discrete alignment actions are implemented by moving data between stacks of Long Short-term Memory (LSTM) blocks, we develop a sequential *spatial* phrase grounding network that we call SeqGROUND. SeqGROUND encodes region proposals and all phrases into two stacks of LSTM cells, along with so-far grounded phrase-region pairings. These LSTM stacks collectively capture the context for the grounding of the next phrase.

Contributions. The contributions of this paper are threefold. First, we propose the notion of contextual phrase grounding, where earlier grounding decisions can inform the latter. Second, we formalize this process in the end-to-end learnable neural architecture we call SeqGROUND. The benefit of this architecture is its ability to sequentially process many-to-many grounding decisions and utilize rich context of prior matches along the way. Third, we show competitive performance both with respect to the prior state-of-the-art and ablation variants of our model. Through ablations we validate the efficacy of sequential grounding as well as individual design choices in our model.

2. Related Work

Localizing phrases in images by performing sequential grounding is related to multiple topics in multi-modal learning. We briefly review the most relevant literature.

Multi-modal Text and Image Tasks. Popular research topics in multi-modal learning include image captioning [19, 28, 45, 52], retrieval of visual content [25], text grounding in images [11, 33, 37, 46] and visual question answering [1, 38, 51]. Most approaches along these lines can be classified as belonging to either (i) joint language-visual embeddings or (ii) encoder-decoder architectures.

The joint *vision-language embeddings* facilitate image/video or caption/sentence retrieval by learning to embed images/videos and sentences into the same space [30, 43, 50, 53]. For example, [15] uses simple kernel CCA and in [8] both images and sentences are mapped into a common semantic *meaning* space defined by object-action-scene triplets. More recent methods directly minimize a pairwise ranking function between positive image-caption pairs and contrastive (non-descriptive) negative pairs; various ranking objective functions have been proposed including max-margin [22] and order-preserving losses [44]. The *encoder-decoder* architectures [43] are similar, but instead attempt to encode images into the embedding space from which a sentence can be decoded.

Of particular relevance is NeuMATCH [7], an architecture for video-sentence alignment, where discrete alignment actions are implemented by moving data between stacks of Long Short-term Memory (LSTM) blocks. We generalize the formulation in [7] to address the spatial grounding of phrases. This requires addition of the spatial proposal mechanism, modifications to the overall architecture in order to allow many-to-many matching, modification to the loss function and a more sophisticated training procedure.

Phrase Grounding. Phrase grounding, a problem addressed in this paper, is defined as spatial localization of the natural language phrase in an image. A number of approaches have been proposed for grounding over the years.

Karpathy *et al.* [20] propose to align sentence fragments and image regions in a subspace. Rohrbach *et al.* [37] propose a method to learn grounding in images by reconstructing a given phrase using an attention mechanism. Fukui *et al.* [11] uses multimodal compact bilinear pooling to represent multimodal features jointly which is then used to predict the best candidate bounding box in a similar way to [37]. Wang *et al.* [47] learns a joint image-text embedding space using a symmetric distance function which is then used to score the bounding boxes to predict the closest to the given phrase. In [46], their embedding network is extended by introducing a similarity network which aggregates multimodal features into a single vector rather than an explicit embedding space. Hu *et al.* [16] proposes a recurrent neural network model to score the candidate boxes using local image descriptors, spatial configurations, and global scene-level context. Plummer *et al.* [33] perform global inference using a wide range of image-text constraints derived from attributes, verbs, prepositions, and pronouns. Yeh *et al.* [55] uses word priors with the combination of segmentation masks, geometric features, and detection scores to select the candidate bounding box. Wang *et al.* [48] proposes a structured matching method which attempts to reflect the semantic relation of phrases onto the visual relations of their corresponding regions without considering the global sentence-level context. Plummer *et al.* [32] proposes to use multiple text-conditioned embeddings in a single end-to-end model with impressive results on Flickr30K Entities dataset [34].

These existing works ground each phrase independently, ignoring the semantic and spatial relations among the phrases and corresponding regions respectively. A notable exception is the approach of Chen *et al.* [3], where a query-guided regression network, designed to regress the rank of candidates phrase-region pairings, is proposed along with a reinforcement learning context policy network for contextual refinement of this ranking. For *referring expression comprehension*, which is closely related to *phrase grounding* problem, [57, 29, 56] introduce taking account of context. Regarding visual data, they consider local context provided by the surrounding objects only. In addition, [29, 56] use textual context with an explicit structure, based on the assumption that referring expressions mention an object in relation with some other object. On the other hand, our method represents visual and textual context in a less structured, but more global, manner which alleviates more explicit assumptions made by other methods. Importantly, unlike [57, 29, 56], it makes use of prior matches through a sequential decision process. In summary, existing approaches perform phrase grounding with two constraints: a region should be matched to no more than one phrase, or a phrase should be matched to no more than one region. Furthermore, most of these approaches consider the local similarities rather taking account both global image-level and sentence-level context. Here we propose an end-to-end differentiable neural architecture that considers all possible sets of bounding boxes to match any phrase in the caption, and vice versa.

3. Approach

We now present our neural architecture for grounding phrases in images. We assume that we need to ground multiple, potentially inter-related, phrases in each image. This is the case for the Flickr30k Entities dataset, where phrases/entities come from sentence parsing. Specifically, we parse the input sentence into a sequence of phrases $\mathcal{P} = \{P_j\}_{j=1...N}$ keeping the sentence order; *i.e.* $j = 1$ is the first phrase and $j = N$ is the last. For a typical sentence in Flickr30k, N is between 1 and 54. The input image I is used to extract region proposals in the form of bounding boxes. These bounding boxes are ordered to form a sequence $\mathcal{B} = \{B_i\}_{i=1...M}$. We discuss the ordering choices, for both $\mathcal{P}$ and $\mathcal{B}$, and their effects in Section 4.3. Our overall task is to ground phrases in the image by matching them to their corresponding bounding boxes, for example, finding a function π that maps an index of the phrase to its corresponding bounding boxes $\langle P_j, B_{\pi(j)} \rangle$. Our method allows many-to-many matching of the aformentioned input sequences. In other words, a single phrase can be grounded to multiple bounding boxes, or multiple phrases of the sentence can be grounded to the same bounding box.

Phrase grounding is a very challenging problem exhibiting the following characteristics. First, image and text are heterogeneous surface forms concealing the true similarity structure. Hence, satisfactory understanding of the entire language and visual content is needed for effective grounding. Second, relationships between phrases and boxes are complex. It is possible (and likely) to have many-to-many matchings and/or unmatched content (due to either lack of precision in the bounding box proposal mechanism or hypothetical linguistic references). Such scenarios need to be accommodated by the grounding algorithm. Third, contextual information that is needed for learning the similarity between phrase-box pairs are scattered over the entire image and the sentence. Therefore, it is important to consider all visual and textual context with a strong representation of their dependencies when making grounding decisions, and to create an end-to-end network, where gradient from grounding decisions can inform content understanding and similarity learning.

The SeqGROUND framework copes with these challenges by casting the problem as one of sequential grounding and explicitly representing the state of the entire decision *workspace*, including the partially grounded input phrases and boxes. The representation employs LSTM recurrent networks for region proposals, sentence phrases, and the previously grounded content, in addition to dense layers for the full image representation. Figure 2 shows the architecture of our framework.

We learn a function that maps the state of workspace Ψ_t to a grounding decision d_{ti} for the bounding box B_i at every time step t, which corresponds to a decision for phrase P_t. The decisions d_{ti} manipulates the content of the LSTM networks, resulting in a new state Ψ_{t+1}. Executing a complete sequence of decisions produces a complete alignment of the input phrases with the bounding boxes. We note, that our model is an extension and generalization of the NeuMATCH framework [7] introduced by Dogan *et al.* Further, there is a clear connection with reinforcement learning and policy gradient methods [41]. While an RL-based formulation maybe a reasonable future extension, here we focus on a fully differentiable supervised learning formulation.

3.1. Language and Visual Encoders

We first create encoders for each phrase and each bounding box produced by a region proposal network (RPN).

Phrase Encoder. The input caption is parsed into phrases $P_1 \dots P_N$, each of which contains a word or a sequence of words, using [2]. We transform each unique phrase into an embedding vector, by performing mean pooling over GloVe [31] features of all its words. This vector is then transformed with three fully connected layers using the ReLU activation function, resulting in the encoded phrase vector p_j for the j^{th} phrase (P_j) of the input sentence.

Visual Encoder. For each proposed bounding box, we ex-

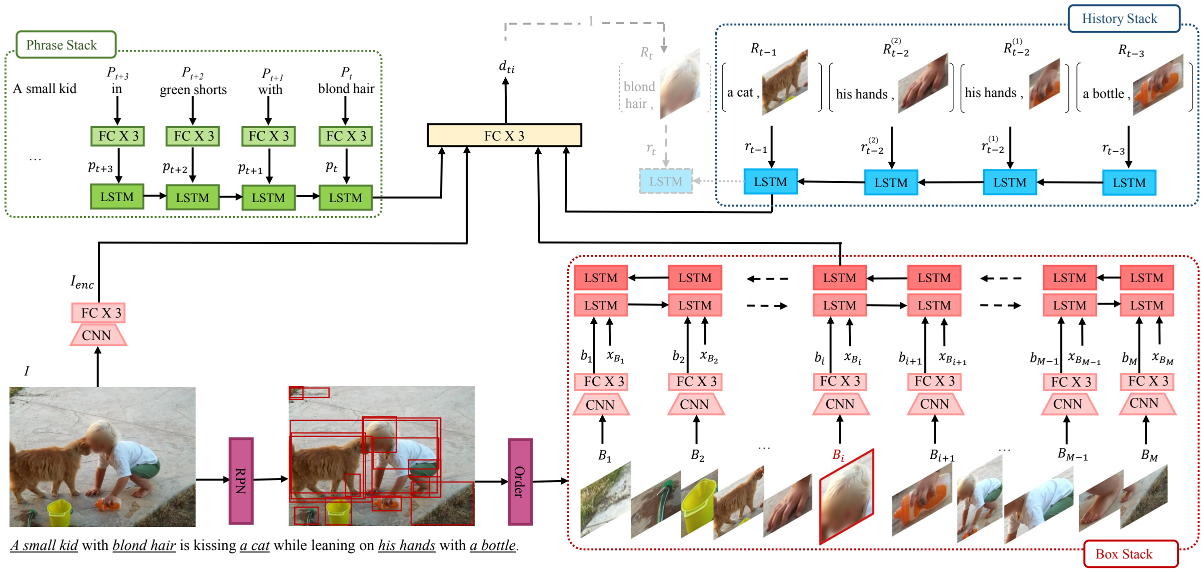


Figure 2: **SeqGROUND neural architecture**. The *phrase stack* contains the sequence of all phrases, not only the noun phrases, yet to be processed in an order and encodes the linguistic dependencies. The *box stack* contains the sequence of bounding boxes that are ordered with respect to their locations in the image. The *history stack* contains the phrase-box pairs that are previously grounded. The grounding decisions for the input phrases are performed sequentially taking into account of the current states of these LSTM stacks in addition to full image representation. The new grounded phrase-box pairs are added to the top of the *history stack*.

tract features using the activation of the first fully connected layer in the VGG-16 network [39], which produces a 4096-dim vector per region. This vector is transformed with three fully connected layers using the ReLU activation function, resulting in the encoded bounding box vector b_i for the i^{th} bounding box (B_i) of the image. The visual encoder is also used to encode the full image I into I_{enc}.

3.2. The Grounding Network

Having the encoded phrases and boxes in the same embedding space, a naive approach for grounding would be maximizing the collective similarity over the grounded phrase-box pairs. However, doing so ignores the spatial structures and relations within the elements of the two sequences, and can lead to degraded performance. SeqGROUND performs grounding by encoding the input sequences and the decision history with stacks of recurrent networks. This implicitly allows the network to take into account all grounded as well as ungrounded proposal regions and phrases as context for the current grounding decision. We show in the experimental section that this leads to a significant boost in performance.

Recurrent Stacks. Considering the input phrases as a temporal sequence, we let the first stack contain the sequence of phrases yet to be processed $P_t, P_{t+1}, \ldots, P_N$, at the time step t. The direction of the stack goes from P_N to P_t, which allows the information to flow from the future phrases to the current phrase. We refer to this LSTM network as the *phrase stack* and denote its hidden state as h^P_t. The input to the LSTM unit is the phrase features in the latent space obtained by the phrase encoder (see Sec. 3.1).

The second stack is a bi-directional LSTM recurrent network that contains the sequence of bounding boxes $B_1, \ldots, B_M$ obtained by the RPN. The boxes are ordered from left to right considering their center on the horizontal axis for the forward network[1]. We refer to this bi-LSTM network as the *box stack* and denote its hidden state for the i^{th} box as h^B_i. The input to the LSTM unit is the concatenation of the box features in the latent space and the normalized location features $[b_i, x_{b_i}]$. Note that the state of the box stack does not change with respect to t. We keep all the boxes in the stack, since a box that is already used to ground a phrase can be used again to grounding another phrase later on.

The third stack is the *history stack*, which contains only the phrases and the boxes that are previously grounded, and places the last grounded phrase-box pair at the top of the stack. We denote this sequence as $R_1, \ldots, R_L$. The information flows from the past to the present. The input to the LSTM unit is the concatenation of the two modalities in the latent space and the location features of the box. When a phrase p_j is grounded to multiple (K)

[1] We experimented with alternative orderings, *e.g.*, max flow computed over pair-wise proposal IoU scores, but saw no appreciable difference in performance. Therefore for cleaner exposition we focus on simpler left-to-right ordering and corresponding results.

boxes $b_{\pi(j)} = b_{(p_j,1)}, \dots, b_{(p_j,K)}$, each grounded phrase-box pair becomes a separate input to the LSTM unit, keeping the spatial order of the boxes. For example, the vector $[p_j, b_{(p_j,1)}, x_{b_{(p_j,1)}}]$ will be the first vector to be pushed to the top of the history stack for the phrase p_j. The last hidden state of the history stack is h^R_{t-1}.

The *phrase stack* and *history stack* both perform encoding using a 2-layer LSTM recurrent network, where the hidden state of the first layer, $h^{(1)}_t$, is fed to the second layer:

$$h^{(1)}_t, c^{(1)}_t = \text{LSTM}(x_t, h^{(1)}_{t-1}, c^{(1)}_{t-1}) \tag{1a}$$

$$h^{(2)}_t, c^{(2)}_t = \text{LSTM}(h^{(1)}_t, h^{(2)}_{t-1}, c^{(2)}_{t-1}) \ , \tag{1b}$$

where $c^{(1)}_t$ and $c^{(2)}_t$ are the memory cells for the two layers, respectively; x_t is the input for time step t.

Image Context. In addition to the recurrent stacks, we also provide the encoded full image I to the network as an additional global context.

Grounding Decision Prediction. At every time step, the state of the three stacks is $\Psi_t = (P_{t^+}, B_t, R_{1^+})$, where we use the shorthand X_{t^+} for the sequence $X_t, X_{t+1}, \dots$ and similarly for X_{t^-}. The LSTM hidden states can approximately represent Ψ_t. Thus, the conditional probability of grounding decision d_{ti}, which represents the decision for bounding box B_i with the phrase P_t is

$$Pr(d_{ti}|\Psi_t) = Pr(d_{ti}|h^P_t, h^B_i, h^R_{t-1}, I_{enc}). \tag{2}$$

In other words, at time step t, a grounding decision is made simultaneously for each box for the phrase at the top of the *phrase stack*. Although it may seem that these decisions are made in parallel independently, the hidden states of the *box stack* encode the relation and dependencies between all the boxes. The above computation is implemented as a sigmoid operation after three fully connected layers on top of the concatenated state $\psi_t = [h^P_t, \{h^B_i\}, h^R_{t-1}, I_{enc}]$. ReLU activation is used between the layers. Further, each positive grounding decision will augment the *history stack*.

In order to ground the entire phrase sequence with the boxes, we apply the chain rule as follows:

$$Pr(D_1, \dots, D_N|\mathcal{P}, \mathcal{B}) = \prod_{t=1}^{N} Pr(D_t|D_{(t-1)^-}, \Psi_t) \tag{3}$$

$$Pr(D_t|\mathcal{P}, \mathcal{B}) = \prod_{i=1}^{M} Pr(d_{ti}|D_{(t-1)^-}, \Psi_t), \tag{4}$$

where D_t represents the set of all grounding decisions over all the boxes for the phrase P_t. The probability can be optimized greedily by always choosing the most probable decisions. The model is trained in a supervised manner. From a ground truth grounding of a box and a phrase sequence, we can easily derive the correct decisions, which are used in training. The training objective is to minimize the overall binary cross-entropy loss caused by the grounding decisions at every time step for each $\langle P_t, B_i \rangle$ with $i = 1, \dots, M$.

Pre-training. As noted in [7], learning a coordinated representation (or similarity measure) between visual and text data, while also optimizing a decision network, is difficult. Thus, we adopt a pairwise pre-training step to coordinate the phrase and visual encoders to achieve a good initialization for subsequent end-to-end training. Note that this is only done for pre-training; the final model is fully differentiable and is fine-tuned end-to-end.

For a ground-truth pair (P_k, B_k), we adopt an asymmetric similarity proposed by [44]

$$F(p_k, b_k) = -||\max(0, b_k - p_k)||^2 \ . \tag{5}$$

This similarity function, F, takes the maximum value 0, when p_k is positioned to the upper right of b_k in the vector space. When that condition is not satisfied, the similarity decreases. In [44], this relative spatial position defines an entailment relation where b_k entails p_k. Here, the intuition is that the image typically contains more information than being described in the text form, so we may consider the text as entailed by the image.

We adopt the following ranking loss objective by randomly sampling a contrastive box B' and a contrastive phrase P' for every ground truth pair. Minimizing the loss function maintains that the similarity of the contrastive pair is below the true pair's by at least the margin α:

$$\begin{aligned}\mathcal{L} = \sum_i \big(&\mathbb{E}_{b' \neq b_k} \max\{0, \alpha - F(b_k, p_k) + F(b', p_k)\} \\ &+ \mathbb{E}_{p' \neq p_k} \max\{0, \alpha - F(b_k, p_k) + F(b_k, p')\}\big)\end{aligned} \tag{6}$$

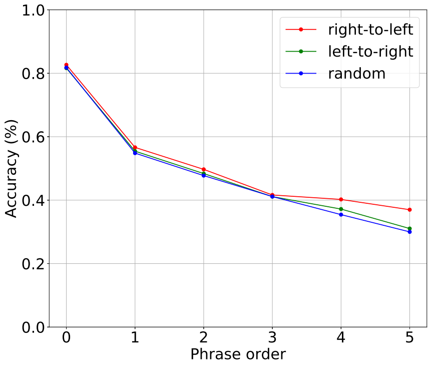
Note the expectations are approximated by sampling.

4. Experiments

4.1. Setup and Training.

We use Faster R-CNN [36] as an underlying bounding box proposal mechanism with ResNet50 as the backbone. The extracted bounding boxes are then sorted from left-to-right by their central x-coordinate to be fed into the Bi-LSTM network of the *box stack*. This way, the objects appearing close tend to be represented closer together, so that the *box stack* can represent the overall context better. Following the prior works (see Tab. 1), we assume that the noun phrases that are to be grounded have already been extracted from the descriptive sentences. We also use the intermediate words of the sentences together with the given noun phrases in the phrase stack to preserve the linguistic structure; this also results in a more complex train/test scenario.

SeqGROUND is trained in two stages that differ in *box stack* input. In the first stage, we only feed the groundtruth



(a)

	Components				
	Visual context	Bounding box	Phrase	History	**Accuracy**
MSB	none	simple	simple	none	43.85
MSBs	none	simple	simple	none	50.90
NH	global	bi-LSTM	LSTM	none	59.55
NI	none	bi-LSTM	LSTM	LSTM	60.34
SPv	global	bi-LSTM	simple	LSTM	57.94
SBv	global	simple	LSTM	LSTM	55.68
SPvBv	global	simple	simple	LSTM	53.75
SBvPvNH	global	simple	simple	none	52.91
SeqGROUND	global	bi-LSTM	LSTM	LSTM	**61.60**

(b)

Figure 3: **The performance of various design choices.** (a) Grounding accuracy versus the ordering of the grounded phrase among the noun phrases of the sentence. Red, green, and blue plots show the performance when the phrases to the LSTM cell are ordered left-to-right (lexical order), right-to-left (reverse lexical order), and randomly, respectively. (b) Grounding accuracy of baselines and ablated models.

instances to the *box stack*, which are coming from the dataset annotation, for an image. The boxes that have the same label as the phrase are considered as positive samples, while the remaining boxes as negative samples. This setup provides an easier phrase grounding task due to the low number of input boxes which are contextually distinct and well-defined without being redundant. Thus, it provides a good initialization for the second stage where we use the box proposals by the RPN.

For the second stage, we map each bounding box, coming from the RPN, to the groundtruth instances with which it has IoU overlap equal to or greater than 0.7, and label them as positive samples for the current phrase. The remaining proposed boxes having IoU overlap less than 0.3 with the groundtruth instances are labeled as negative samples for that phrase. The labeled positive and negative samples are sorted and then fed into the Bi-LSTM network. It is possible to optimize for the loss function of all labeled boxes, but this will bias towards negative samples as they dominate. Instead, we randomly sample negative samples that contribute to the loss function in a batch, where the sampled positive and negative boxes have a ratio of 1:3. If the number of negative samples within a batch is not enough, we let all the samples in that batch contribute to the loss. In this way, the spatial context and dependencies are represented without gaps by the Bi-LSTM unit of the *box stack*, while preventing biasing towards negative grounding decisions. After the second stage of training, we adopt the standard hard negative mining method [9, 40] with a single pass on each training sample.

At test time, we use all the proposed boxes to feed them to the *box stack* after ordering them with respect to their locations. When multiple boxes are grounded to the same phrase, we apply non-maximum suppression with an IoU overlap threshold of 0.3, which is tuned on the validation set. In this way, multiple box results for the same instance of a phrase are discarded, while the boxes for different instances of the same phrase are kept. More implementation details are available in the supplementary material.

4.2. Datasets and Metrics

We evaluate our approach on the Flickr30K Entities dataset [34] which contains $31,783$ images, each annotated with five sentences. For each sentence, the noun phrases are provided with their corresponding bounding boxes in the image. We use the same training/validation/test split as the prior work, which provides $1,000$ images for validation, $1,000$ for testing, and $29,783$ images for training. It is important to note that a single phrase can have multiple groundtruth boxes, while a single box can match multiple phrases within the same sentence. Consistent with the prior work, we evaluate SeqGROUND with the ground truth bounding boxes. If multiple boxes are associated with a phrase, we represent the phrase as the union of all its boxes on the image plane. Following the prior work, successful grounding of a phrase requires predicted area to have at least 0.5 IoU (intersection over union) with the groundtruth area. Based on this criteria, our measure of performance is grounding *accuracy*, which is the ratio of correctly grounded noun phrases.

4.3. Baselines and Ablation Studies

In order to understand the benefits of the individual components of our model, we perform an ablation study where certain stacks are either removed or modified. The model *NH* lacks the *history stack* where the previously grounded phrase-box pairs do not affect the decisions for the upcoming phrases in a sentence. The model *NI* lacks the full image context where the only visual information to the framework is the *box stack*. The model *SBv* (simple box vector) lacks the bi-LSTM network for the boxes, and direclty uses the encoded box features coming from the triple fully connected layers in Figure 2. In this way, the decision for

Method	Accuracy
SMPL [48]	42.08
NonlinearSP [47]	43.89
GroundeR [37]	47.81
MCB [11]	48.69
RtP [34]	50.89
Similarity Network [46]	51.05
RPN+QRN [3]	53.48
IGOP [55]	53.97
SPC+PPC [33]	55.49
SS+QRN [3]	55.99
CITE [32]	59.27
SeqGROUND	**61.60**

Table 1: **Phrase grounding accuracy** (in percentage) of the state-of-the-art methods on the Flickr30k Entities dataset.

a phrase-box pair is made independently of the other box candidates. The model *SPv* (simple phrase vector) lacks the LSTM network for the *phrase stack* and directly uses the encoded phrase features coming from the triple fully connected layers in Figure 2. In this design, the framework is not aware of the upcoming phrases so that the decision for a phrase-box pair is made without the linguistic relations. Similarly, *SPvBv* lacks the bi-LSTM and LSTM networks for the box and phrase stacks, respectively. Moreover, *SPvBvNH* lacks the history module as an addition.

Moreover, we created a baseline that performs phrase grounding in a non-sequential way by picking the most similar bounding box in the joint embedding space. To encode the phrases and boxes, we used the same phrase-visual encoders that were pre-trained in Section 3.2. For each image-sentence input, we created a similarity matrix for all possible phrase-box pairs using the similarity function 5. Using this matrix, the phrases were grounded to the most similar box and boxes for the models *MSB* and *MSBs*, respectively.

Table 3b shows the performance of the six ablated models and two baselines on the Flickr30K Entities dataset. All these models perform substantially worse than the complete model of SeqGROUND. This confirms our intuition that knowing the global context for both visual and textual data, in addition to history and future, plays an important role in phrase grounding. We conclude that each stack contributes to our full model's superior performance.

Phrase Ordering. We consider several ways of ordering the phrases of a sentence.

1. Left-to-Right: The network grounds the phrases in lexical order, starting from the first phrase of the sentence.
2. Right-to-Left: The network grounds the phrases in reverse lexical order, starting from the last phrase.
3. Random: We randomly order the phrases, and keep the ordering fixed for all of the training.

At test time, the phrases are ordered in the same order as the corresponding design's training time. The grounding accuracy with respect to the phrase's order among the noun phrases of the sentence is shown in Figure 3a for different ordering options. For all ordering options, the accuracy for the first phrase is significantly higher than the others. This is due to the fact that the first phrases usually belong to the category of *people* or *animals* which have significantly more samples in the dataset. Moreover, the candidate boxes from RPN are more accurate in proposing boxes for these categories which provides easier detection. The grounding accuracy drops towards the last phrases, which usually belong to the categories that have less samples in the dataset. Ordering the phrases right-to-left boosts the performance slightly for the last phrases of the sentence, since they are the first ones to be grounded. In this way, these hard-to-ground phrases are not a subject of a possible error cumulation in the *history stack*.

Unguided Testing. SeqGROUND does not necessarily need to be given phrases to ground. Due to its sequential nature, it scans through all the phrases in the sentences, selected phrases or not, and makes decisions which of those to ground and where (see Fig. 4). The network implicitly learns to distinguish entities to-be-grounded during training. This is a more complex scenario than addressed by prior works, which only focus on phrases that implicitly have groundings. The results in Table 1, 2, and Figure 4 are obtained via unguided testing, which is a key property of our method.

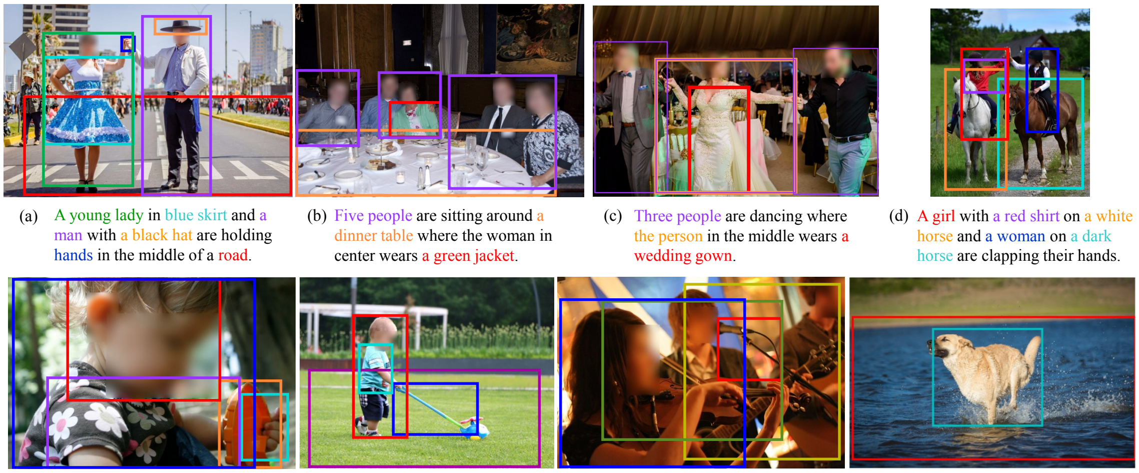
4.4. Results

We report the performance of SeqGROUND on the Flickr30K Entities dataset, and compare it with the state-of-the-art methods[3] in Table 1. SeqGROUND is the top ranked method in the list, improving the overall grounding accuracy by 2.33% to 12.91% by performing phrase grounding as a sequential and contextual process, compared to the prior work. For a fair comparison, all these methods use a fixed RPN to obtain the candidate boxes and represent them in features that are not tuned on the Flickr30K Entities dataset. We believe that using an additional conditional embedding unit as in [32], and the integration of a proposal generation network with a spatial regression that is tuned on Flickr30K Entities as in [3] should improve the overall result even more. Table 2 shows the phrase grounding performance with respect to the coarse categories in Flickr30K Entitites dataset. Competing results are directly taken from the respective papers, if applicable.

[3] Performance on this task can be further improved by using Flickr30K-tuned features to represent the image regions, with the best result of 61.89% achieved by CITE [32]. Futhermore, the use of an integrated proposal generation network to learn regression over Flickr30K Entities improves the result up to 65.14% as achieved by [3].

Method	people	clothing	body parts	animals	vehicles	instruments	scene	other
SMPL [48]	57.89	34.61	15.87	55.98	52.25	23.46	34.22	26.23
GroundeR [37]	61.00	38.12	10.33	62.55	68.75	36.42	58.18	29.08
RtP [34]	64.73	46.88	17.21	65.83	68.72	37.65	51.39	31.77
IGOP [55]	68.71	56.83	19.50	70.07	73.72	39.50	60.38	32.45
SPC+PPC [33]	71.69	50.95	25.24	76.23	66.50	35.80	51.51	35.98
CITE [32]	73.20	52.34	30.59	76.25	75.75	48.15	55.64	42.83
SeqGROUND	76.02	56.94	26.18	75.56	66.00	39.36	68.69	40.60

Table 2: **Comparison of phrase grounding accuracy** (in percentage) over coarse categories on Flickr30K dataset.



(a) A young lady in blue skirt and a man with a black hat are holding hands in the middle of a road.

(b) Five people are sitting around a dinner table where the woman in center wears a green jacket.

(c) Three people are dancing where the person in the middle wears a wedding gown.

(d) A girl with a red shirt on a white horse and a woman on a dark horse are clapping their hands.

(e) A baby with blond hair in flower patterned shirt holding an orange toy in her hand.

(f) A toddler in a blue shirt is steering his toy on a grass field.

(g) A young woman is playing a violin while a young man is singing to a microphone.

(h) A white dog is running over the water.

Figure 4: **Sample phrase grounding results obtained by SeqGROUND[2]**. The colored bounding boxes show the predicted grounding of the phrases in the same color. See text for discussion.

We show some qualitative results in Figure 4 to highlight the capabilities of our method in challenging scenarios. In (a) and (e), we see a successful grounding of long sequence of phrases, note the correct grounding of *hands* in (a) despite other *hands* candidates. In (b), phrases are correctly grounded to multiple boxes, instead of one large single box for *five people* which would contain mostly the *dinner table*. Likewise, (c) shows an example where a single box is used to ground multiple phrases, *three people* and *the person* which are positioned far apart. Phrase grounding with many-to-many matching is one of the distinguishing properties of SeqGROUND, which is partially or completely missing in most of the competing methods. In (d), SeqGROUND could distinguish which boxes to ground the phrases *a girl* and *a woman*, suppressing the other candidates despite their similar context. We believe this is possibly due to SeqGROUND's ability to perform in a sequential way where it consders the global image and text context. As an intuitive example, the performed grounding starts by matching *a dark horse* to the correct box. Encoding this grounded pair and the overall contextual information, it grounds *a woman* to the correct box, which is just above *a dark horse*, instead of getting confused by the box that has *A girl*. At the decision time for *a woman*, the *phrase stack* encodes the future information, which is *a girl* should have a *red shirt* and should be on *a white horse*. Taking account of this information likely has led SeqGROUND to eliminate the box for *a girl* at the decision time for *a woman*.

All these images, and more in the supplementary material, show state-of-the-art performance of SeqGROUND due to its contextual and sequential nature.

5. Conclusions

In this paper, we proposed an end-to-end trainable Sequential Grounding Network (SeqGROUND) that formulates grounding of multiple phrases as a sequential and contextual process. SeqGROUND encodes region proposals, and all phrases into two stacks of LSTM cells along with the partially grounded phrase-region pairs to perform the grounding decision for the next phrase. Results on the Flickr30K Entities benchmark dataset and ablations studies show significant improvements of this model over more traditional grounding approaches.

[3] Due to copyright issues of Flickr30K Entities dataset, we are not allowed to show images from it. Instead, we created similar content with *public domain* images, and blurred faces due to privacy concerns.

References

[1] Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C. Lawrence Zitnick, and Devi Parikh. VQA: Visual question answering. In *ICCV*, pages 2425–2433, 2015.

[2] Danqi Chen and Christopher Manning. A fast and accurate dependency parser using neural networks. In *EMNLP*, pages 740–750, 2014.

[3] Kan Chen, Rama Kovvuri, and Ram Nevatia. Query-guided regression network with context policy for phrase grounding. In *ICCV*, 2017.

[4] L.-C. Chen, Y. Yang, J. Wang, W. Xu, and A. L. Yuille. Attention to scale: Scale-aware semantic image segmentation. In *CVPR*, 2016.

[5] Xinlei Chen and Abhinav Gupta. Spatial memory for context reasoning in object detection. In *ICCV*, 2017.

[6] Xinlei Chen, Li-Jia Li, Li Fei-Fei, and Abhinav Gupta. Iterative visual reasoning beyond convolutions. In *CVPR*, 2018.

[7] Pelin Dogan, Boyang Li, Leonid Sigal, and Markus Gross. A neural multi-sequence alignment technique (neumatch). In *CVPR*, pages 8749–8758, 2018.

[8] Ali Farhadi, Mohsen Hejrati, Mohammad Amin Sadeghi, Peter Young, Cyrus Rashtchian, Julia Hockenmaier, and David Forsyth. Every picture tells a story: Generating sentences from images. In *ECCV*, pages 15–29, 2010.

[9] Pedro F Felzenszwalb, Ross B Girshick, David McAllester, and Deva Ramanan. Object detection with discriminatively trained part-based models. *IEEE transactions on pattern analysis and machine intelligence*, 32(9):1627–1645, 2010.

[10] A. Fukui, D.H. Park, D. Yang, A. Rohrbach, T. Darrell, and M. Rohrbach. Multi- modal compact bilinear pooling for visual question answering and visual grounding. In *EMNLP*, 2016.

[11] Akira Fukui, Dong Huk Park, Daylen Yang, Anna Rohrbach, Trevor Darrell, and Marcus Rohrbach. Multimodal compact bilinear pooling for visual question answering and visual grounding. *arXiv preprint arXiv:1606.01847*, 2016.

[12] A. Gordo, J. Almazan, J. Revaud, and D. Larlus. Deep image retrieval: Learning global representations for image search. In *ECCV*, 2016.

[13] Kirk Hazen. *An Introduction to Language*. Wiley Blackwell, 2014.

[14] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016.

[15] Micah Hodosh, Peter Young, and Julia Hockenmaier. Framing image description as a ranking task: Data, models and evaluation metrics. *Journal of Artificial Intelligence Research*, 47:853–899, 2013.

[16] Ronghang Hu, Huazhe Xu, Marcus Rohrbach, Jiashi Feng, Kate Saenko, and Trevor Darrell. Natural language object retrieval. In *CVPR*, pages 4555–4564, 2016.

[17] J. Johnson, A. Karpathy, and L. Fei-Fei. Densecap: Fully convolutional localization networks for dense captioning. In *CVPR*, 2016.

[18] A. Karpathy and L. Fei-Fei. Deep visual-semantic alignments for generating image descriptions. In *CVPR*, 2015.

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

[20] Andrej Karpathy, Armand Joulin, and Li F Fei-Fei. Deep fragment embeddings for bidirectional image sentence mapping. In *Advances in neural information processing systems*, pages 1889–1897, 2014.

[21] S. Kazemzadeh, V. Ordonez, M. Matten, and T. L. Berg. Referitgame: Referring to objects in photographs of natural scenes. In *EMNLP*, 2014.

[22] Ryan Kiros, Ruslan Salakhutdinov, and Richard S Zemel. Unifying visual-semantic embeddings with multimodal neural language models. *Transactions of the Association for Computational Linguistics*, 2014.

[23] R. Krishna, Y. Zhu, O. Groth, J. Johnson, K. Hata, J. Kravitz, S. Chen, Y. Kalantidis, L.-J. Li, D. A. Shamma, M. Bernstein, and L. Fei-Fei. Visual genome: Connecting language and vision using crowdsourced dense image annotations. In *IJCV*, 2017.

[24] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[25] Dahua Lin, Sanja Fidler, Chen Kong, and Raquel Urtasun. Visual semantic search: Retrieving videos via complex textual queries. In *CVPR*, pages 2657–2664, 2014.

[26] R. Luo and G. Shakhnarovich. Comprehension-guided referring expressions. In *CVPR*, 2017.

[27] Junhua Mao, Jonathan Huang, Alexander Toshev, Oana Camburu, Alan Yuille, and Kevin Murphy. Generation and comprehension of unambiguous object descriptions. In *CVPR*, 2016.

[28] Junhua Mao, Wei Xu, Yi Yang, Jiang Wang, Zhiheng Huang, and Alan Yuille. Deep captioning with multimodal recurrent neural networks (m-rnn). *arXiv preprint*, 2014.

[29] Varun K Nagaraja, Vlad I Morariu, and Larry S Davis. Modeling context between objects for referring expression understanding. In *European Conference on Computer Vision*, pages 792–807. Springer, 2016.

[30] Yingwei Pan, Tao Mei, Ting Yao, Houqiang Li, and Yong Rui. Jointly modeling embedding and translation to bridge video and language. In *CVPR*, pages 4594–4602, 2016.

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

[32] Bryan A Plummer, Paige Kordas, M Hadi Kiapour, Shuai Zheng, Robinson Piramuthu, and Svetlana Lazebnik. Conditional image-text embedding networks. In *ECCV*, 2018.

[33] Bryan A Plummer, Arun Mallya, Christopher M Cervantes, Julia Hockenmaier, and Svetlana Lazebnik. Phrase localization and visual relationship detection with comprehensive image-language cues. In *ICCV*, 2017.

[34] Bryan A Plummer, Liwei Wang, Chris M Cervantes, Juan C Caicedo, Julia Hockenmaier, and Svetlana Lazebnik. Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models. In *ICCV*, pages 2641–2649, 2015.

[35] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *CVPR*, June 2016.

[36] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, pages 91–99, 2015.

[37] Anna Rohrbach, Marcus Rohrbach, Ronghang Hu, Trevor Darrell, and Bernt Schiele. Grounding of textual phrases in images by reconstruction. In *ECCV*, pages 817–834. Springer, 2016.

[38] Fereshteh Sadeghi, Santosh K Kumar Divvala, and Ali Farhadi. Viske: Visual knowledge extraction and question answering by visual verification of relation phrases. In *CVPR*, pages 1456–1464, 2015.

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

[40] K-K Sung and Tomaso Poggio. Example-based learning for view-based human face detection. *IEEE Transactions on pattern analysis and machine intelligence*, 20(1):39–51, 1998.

[41] Richard S Sutton and Andrew G Barto. Reinforcement learning: An introduction. 2011.

[42] T. Tommasi, A. Mallya, B.A. Plummer, S. Lazebnik, A.C. Berg, and T.L. Berg. Solving visual madlibs with multiple cues. In *BMVC*, 2016.

[43] Atousa Torabi, Niket Tandon, and Leonid Sigal. Learning language-visual embedding for movie understanding with natural-language. *arXiv preprint arXiv:1609.08124*, 2016.

[44] Ivan Vendrov, Ryan Kiros, Sanja Fidler, and Raquel Urtasun. Order-embeddings of images and language. *arXiv preprint arXiv:1511.06361*, 2015.

[45] Oriol Vinyals, Alexander Toshev, Samy Bengio, and Dumitru Erhan. Show and tell: A neural image caption generator. In *CVPR*, pages 3156–3164, 2015.

[46] Liwei Wang, Yin Li, Jing Huang, and Svetlana Lazebnik. Learning two-branch neural networks for image-text matching tasks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2018.

[47] Liwei Wang, Yin Li, and Svetlana Lazebnik. Learning deep structure-preserving image-text embeddings. In *CVPR*, pages 5005–5013, 2016.

[48] Mingzhe Wang, Mahmoud Azab, Noriyuki Kojima, Rada Mihalcea, and Jia Deng. Structured matching for phrase localization. In *ECCV*, pages 696–711. Springer, 2016.

[49] F. Xiao, L. Sigal, and Y. J. Lee. Weakly-supervised visual grounding of phrases with linguistic structures. In *CVPR*, 2017.

[50] B. Xu, Y. Fu, Y. G. Jiang, B. Li, and L. Sigal. Heterogeneous knowledge transfer in video emotion recognition, attribution and summarization. *IEEE Transactions on Affective Computing*, 2017.

[51] Huijuan Xu and Kate Saenko. Ask, attend and answer: Exploring question-guided spatial attention for visual question answering. In *ECCV*, pages 451–466, 2016.

[52] Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhudinov, Rich Zemel, and Yoshua Bengio. Show, attend and tell: Neural image caption generation with visual attention. In *ICML*, pages 2048–2057, 2015.

[53] Ran Xu, Caiming Xiong, Wei Chen, and Jason J Corso. Jointly modeling deep video and compositional text to bridge vision and language in a unified framework. In *AAAI*, volume 5, page 6, 2015.

[54] Jianwei Yang, Jiasen Lu, Stefan Lee, Dhruv Batra, and Devi Parikh. Graph r-cnn for scene graph generation. In *ECCV*, 2018.

[55] Raymond Yeh, Jinjun Xiong, Wen-Mei Hwu, Minh Do, and Alexander Schwing. Interpretable and globally optimal prediction for textual grounding using image concepts. In *Advances in Neural Information Processing Systems*, pages 1912–1922, 2017.

[56] Licheng Yu, Zhe Lin, Xiaohui Shen, Jimei Yang, Xin Lu, Mohit Bansal, and Tamara L Berg. Mattnet: Modular attention network for referring expression comprehension. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1307–1315, 2018.

[57] Licheng Yu, Patrick Poirson, Shan Yang, Alexander C Berg, and Tamara L Berg. Modeling context in referring expressions. In *European Conference on Computer Vision*, pages 69–85. Springer, 2016.

[58] Y. Zhang, L. Yuan, Y. Guo, Z. He, I.A. Huang, and H. Lee. Discriminative bimodal networks for visual localization and detection with natural language queries. In *CVPR*, 2017.