Decoding Language Spatial Relations to 2D Spatial Arrangements

Gorjan Radevski[1], Guillem Collell[2], Marie-Francine Moens[2], Tinne Tuytelaars[1]
[1]Department of Electrical Engineering (ESAT)
[2]Department of Computer Science (CS)
KU Leuven
{gorjan.radevski, tinne.tuytelaars}@esat.kuleuven.be
{guillem.collell, sien.moens}@cs.kuleuven.be

Abstract

We address the problem of multimodal spatial understanding by decoding a set of language-expressed spatial relations to a set of 2D spatial arrangements in a multi-object and multi-relationship setting. We frame the task as arranging a scene of clip-arts given a textual description. We propose a simple and effective model architecture SPATIAL-REASONING BERT (SR-BERT), trained to decode text to 2D spatial arrangements in a non-autoregressive manner. SR-BERT can decode both explicit and implicit language to 2D spatial arrangements, generalizes to out-of-sample data to a reasonable extent and can generate complete abstract scenes if paired with a clip-arts predictor. Finally, we qualitatively evaluate our method with a user study, validating that our generated spatial arrangements align with human expectation.

1 Introduction

Spatial understanding is a problem of paramount importance to both the vision and the language community. For a machine learning model to be able to reason about the spatial domain w.r.t. another modality (language, vision, etc.), it should incorporate common-sense spatial knowledge, which is often obvious to humans, yet hard to grasp by machines. If the spatial relations are expressed in language, such common sense knowledge can be hidden within, e.g., "mike and jenny see a duck" (Figure 1) – meaning that both "Mike" and "Jenny" should be facing the "duck". The complexity of the problem increases tremendously when there is no limit on the number/type of objects and relationships – the de-facto setting in computer graphics, video games, 3D modelling, etc. Communicating spatial relations via language is intuitive for humans, while arranging objects in a multi-dimensional space is tedious. Therefore, building

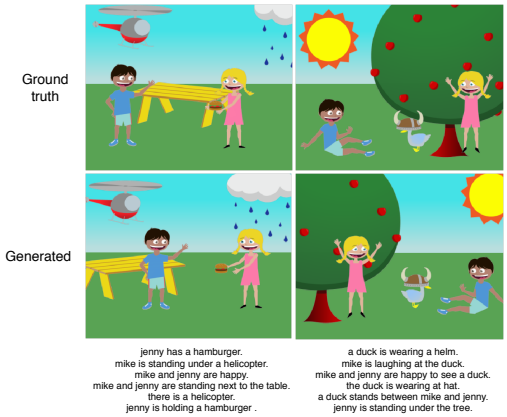

Figure 1: Given a set of clip-arts and a textual description of a scene, including both implicit as well as explicit language, our method automatically generates a reasonable spatial arrangement.

models that exhibit spatial understanding is a key step towards automation – aiding humans in the repetitive time-consuming tasks.

To date, multiple methods that explicitly investigate spatial reasoning in a multidimensional space have been proposed. However, the main limitations are: (i) scene environments with strong priors on (relative) object placements (e.g., indoor home environments); (Chang et al., 2017; Fisher et al., 2012; Choi et al., 2013; Xu et al., 2013; Jiang et al., 2012; Chang et al., 2014; Kermani et al., 2016); (ii) modelling only pairwise relationships (i.e., two objects and a single relationship) (Dan et al., 2020a; Collell et al., 2018); (iii) not using natural language descriptions of scenes, but only structured language (Collell et al., 2018; Dan et al., 2020b).

In this paper we address the three limitations above by introducing a model that analyzes all available textual and visual data jointly. We formally frame our research problem as: "Given a set of discretely encoded clip-arts (people, objects, etc.), and a textual description of a scene, what

Findings of the Association for Computational Linguistics: EMNLP 2020, pages 4549–4560
November 16 - 20, 2020.

is the best positioning of the clip-arts that corresponds to the spatial relations implied by the text?". Our approach is based on a large pre-trained language model – BERT (Devlin et al., 2018), adapted to jointly process multi-modal data with distinct positional encoding. We introduce SR-BERT, a model that explicitly focuses on the spatial relations – decoding the language cues to 2D spatial arrangements, achieved by masking the information related to the spatial arrangements during training. We build on the methods of Ghazvininejad et al. (2019a); Kasai et al. (2020); Wang and Cho (2019); Lee et al. (2018), initially proposed for non-autoregressive text decoding, in our case specifically adapted to iteratively mask-out and predict the spatial arrangements of the objects of interest. Inspired by Lawrence et al. (2019) we develop distinct ways of imposing a decoding order, tailored for generating spatial arrangements from language.

We perform ad-hoc experiments to gain insights in three main research questions: (RQ 1) Can we decode a set of language spatial relations to the 2D space without imposing constraints on the number and type of objects and relationships? (RQ 2) Does the model merely exploit dataset bias to generate arrangements or does it acquire understanding of the language and the spatial domain? (RQ 3) Is the model able to interpret only explicit spatial relationships (e.g., on, above) or can it cope with implicit ones as well (e.g., wearing, eating, etc.)? We release the code, data and trained models[1].

2 Dataset

We use the Abstract Scenes dataset (Zitnick and Parikh, 2013) which consists of 10.020 scenes of clip-arts, together with ∼6 sentences for each scene, describing the scene content and spatial relations between them. The clip-arts belong to 7 distinct groups, namely objects in the sky, large elements, people, animals, clothing, food and toys. The scenes are organized in 1002 semantically different sets, where scenes within a particular set are generated from the same core-scene description. After removing empty scenes, from each of the 1002 sets we allocate one scene for testing, one for model selection and we keep the rest for training.[2] That leaves us with 1002 scenes in the test and validation set respectively, and 7989 scenes in the training set. The maximum number of clip-arts in a scene is 17 while the minimum and median are 6. The total number of unique clip-arts in the dataset is 126.

3 Methods

The main building block for all our models is BERT (Devlin et al., 2018). In particular, we present SR-BERT, a BERT variant based on a pre-trained $\text{BERT}_{\text{BASE}}$. Compared with existing BERT architectures (Sun et al., 2019; Tan and Bansal, 2019; Chen et al., 2019; Su et al., 2019; Lu et al., 2019; Li et al., 2019b), with SR-BERT our contributions are two-fold: (i) We alter the input-embedding module to process two discrete modalities with a different positional encoding — sequential and spatial. (ii) We design a novel training method – Masked Position Modelling, where we iteratively mask and predict the positional encoding of the input tokens.

3.1 BERT revisited

In BERT, the input sequence is tokenized using WordPiece tokenization (Wu et al., 2016) and encoded in token indices $\{w^1, ..., w^N\}$ with a `[CLS]` token index prepended at the start and a `[SEP]` token index appended at the end. Then, a token embedding vector, a token index and a token type embedding vector for each word are summed, and subsequently layer-normalization (Ba et al., 2016) and dropout (Srivastava et al., 2014) are applied. The rest of the architecture resembles the Transformer model of Vaswani et al. (2017). The essence of BERT's bi-directionality is the pre-training method - Masked Language Modelling (MLM) which we explain in section 3.3. For a detailed description we refer to Devlin et al. (2018).

3.2 SR-BERT

In order to enable BERT to handle two modalities with different positional encoding, we keep the language embedding module as described in section 3.1 before the addition of the 3 embeddings and append the output of a specialized spatial embedding module. Here, each clip-art is encoded with a unique index c_i and the spatial encoding is a $[c_x, c_y]$ coordinate pair and a binary indicator c_o of the clip-art orientation (left or right). Therefore, the spatial embedding module consists of 4 separate trainable layers: CEmbed – for obtaining a clip-art embedding c_e, XEmbed, YEmbed – for obtaining

[1] https://github.com/gorjanradevski/sr-bert

[2] We do the data-split to retain as much information as possible within the train-validation-test splits. We also include an experiment with a random split in appendix D.

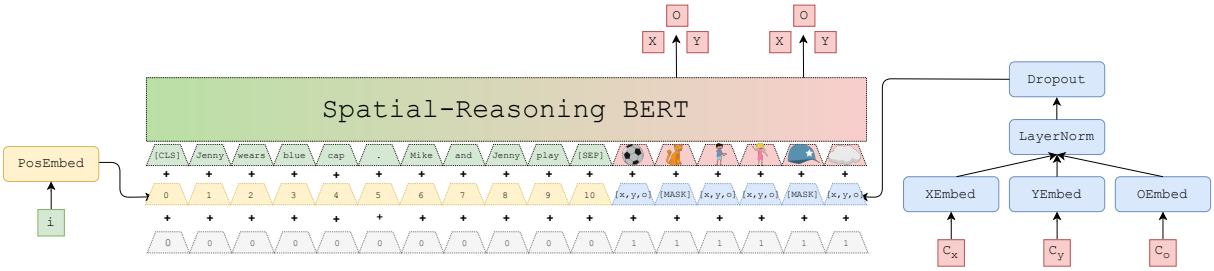

Figure 2: The SR-BERT backbone architecture with the text position embedding module as per BERT – Left (Yellow), clip-art spatial embedding module, which is novel in SR-BERT – Right (Blue). The blue [MASK] elements are the masked spatial positions, which the model learns to predict during training. During inference, all blue elements (the spatial encoding of the clip-arts) are masked, and the model non-autoregressively decodes them.

a spatial embedding $[x_e, y_e]$ for the x and y axis respectively, and OEmbed – for obtaining the spatial embedding o_e of the orientation c_o. These are combined in the final spatial embedding s_e. Finally, we obtain a token type embedding t^c_e. Consequently, for a single element c_i we compute:

$$\begin{aligned} c_e &= \text{CEmbed}(c_i), \quad x_e = \text{XEmbed}(c_x) \\ y_e &= \text{YEmbed}(c_y), \quad o_e = \text{OEmbed}(c_o) \\ s_e &= \text{Drop}(\text{LayerNorm}(\lambda(x_e + y_e + o_e))) \\ t^c_e &= \text{TokenTypeEmbed}(c_i) \end{aligned} \tag{1}$$

where λ is a scaling factor. Finally, we concatenate the word embeddings w_e with the clip-art embeddings c_e, the word positional embeddings p_e with the spatial embeddings s_e, the token type embeddings for the language t^w_e and spatial parts t^c_e, and apply layer-normalization and dropout:

$$\begin{aligned} wc &= \text{Concat}(w_e, c_e) \\ ps &= \text{Concat}(p_e, s_e) \\ tt &= \text{Concat}(t^w_e, t^c_e) \\ e &= \text{Drop}(\text{LayerNorm}(wc + ps + tt)) \end{aligned} \tag{2}$$

where e is the final input embedding. We keep the rest of the model identical to (Devlin et al., 2018), and re-use the pre-trained $\text{BERT}_{\text{BASE}}$ modules. Figure 2 illustrates our model's backbone. On top, we append modelling heads that consist of two linear layers, with a GELU (Hendrycks and Gimpel, 2016) and a layer-normalization between them:

$$h = \text{Linear}(\text{LayerNorm}(\sigma(\text{Linear}(x)))) \tag{3}$$

where x is the hidden representation of a single element. We create a continuous and a discrete model[3] variant where both output a probability for the object orientation o:

$$o_{out} = \text{softmax}(h_o) \tag{4}$$

and minimize a cross-entropy loss $\mathcal{L}_o$ during training. Then, the continuous model generates an $[x, y]$ pair for the clip-art position within the $[0, 500]$ and $[0, 400]$ range respectively, by applying sigmoid on top of two modelling heads h_x and h_y, subsequently multiplied by x_{max} and y_{max}:

$$\begin{aligned} x_{out} &= \sigma(h_x) * x_{max} \\ y_{out} &= \sigma(h_y) * y_{max} \end{aligned} \tag{5}$$

and performs a direct optimization of the similarity measures during training by minimizing a sum of all individual losses: $\mathcal{L} = \mathcal{L}_{abs} + \mathcal{L}_{rel} + \mathcal{L}_o$[4] (explained in section 4.1). When regressing a multimodal function, there is the risk of the model converging to the mean in between modes. To overcome this, we develop a discrete model that outputs a probability distribution over the quantized x and y axis (explained in section 3.3) by applying softmax on top of two modelling heads h_x and h_y:

$$\begin{aligned} x_{out} &= \text{softmax}(h_x) \\ y_{out} &= \text{softmax}(h_y) \end{aligned} \tag{6}$$

and train the model by minimizing the sum of the individual per-axis cross-entropy losses $\mathcal{L}_x$, $\mathcal{L}_y$ together with the orientation loss $\mathcal{L}_o$: $\mathcal{L} = \mathcal{L}_x + \mathcal{L}_y + \mathcal{L}_o$. With the discrete model, we use the arguments of the maxima over the x and y axis during inference to get a clip-art scene location.

3.3 Masked Position Modelling (MPM)

Originally, BERT is trained by reconstructing the ground truth sentence given a corrupted one as input, where 15% of the tokens are replaced with a

[3] Named according to the $[x, y]$ value they predict.

[4] We remove the Gaussian kernel to make the similarity measures into distance functions that can be minimized.

[MASK] token 80% of the time, a random token 10% of the time, or unchanged 10% of the time. Then, BERT outputs a probability distribution for each token, while the loss is computed over the masked tokens. To conceptually preserve BERT's input-embedding module and explicitly encode a masked clip-art position, we encode the spatial position with a discrete set of values. In our use-case, the ranges are $[0, 500]$ for x, $[0, 400]$ for y and $[0, 1]$ for o. Due to the data size (∼8000 scenes for training with a median of 6 clip-arts per scene), having $500 \times 400 \times 2$ spatial combinations is unfeasible to learn. To overcome that, we quantize the values of x and y in intervals of 20, yielding a range of $[0, 25]$ unique values for x and $[0, 20]$ for y.

In SR-BERT, we adjust the masking objective to decode spatial representations, i.e., *instead of masking the clip-art tokens, we mask the spatial encoding tokens*. Namely, given a set of sentences and set of clip-arts with their spatial position $[x, y, o]$, we train the model such that when the scene elements' position is corrupted, the model learns to rely on the relations from the sentences to reconstruct the original layout. Thus, as per Devlin et al. (2018) we mask 15% of the scene elements' spatial positions during training. Then, 80% of the time the $[x, y, o]$ spatial encoding is replaced with [[MASK],[MASK],[MASK]] tokens, 10% of the time with a random $[x, y]$ position and random o orientation, and 10% of the time we keep them the same.

During training, we employ data augmentation techniques (see appendix B), specifically adapted to fit within the training objective we propose.

3.4 Non-autoregressive decoding of spatial arrangements

Despite the non-sequential nature of the 2D space, we hypothesise that decoding spatial arrangements without following any particular order or in a single-step manner is undesirable. In contrast with the left-to-right sequential order in written English, decoding a set of spatial relationships does not have a pre-defined order and one must consider all pairwise relative locations of the objects. Consequently, the model either has to learn the decoding order itself, or the decoding order has to be injected manually. Here, we adopt non-autoregressive decoding based on mask-predict (Ghazvininejad et al., 2019b) to convert language spatial relations into 2D spatial arrangements. By using a transformer architecture with mask-predict, we can easily derive decoding strategies where the model considers all relations between the objects and learns the decoding order. Specifically, we assess five decoding strategies inspired by Lawrence et al. (2019):

(i) **Single-step (SS)**: The spatial arrangements for all objects in the scene are generated in a single step. (ii) **Random-order (RO)**: Following no particular order, we generate the spatial arrangements one by one. (iii) **Human-order (HO)**: We inject domain knowledge in the generation process, and generate spatial arrangements following an intuitive order: objects in the sky, large elements, people, animals, clothing, food and toys. (iv) **Highest-confidence (HC) - discrete**: We maintain a fixed beam of 3 hypotheses for the scene elements that exhibit the highest joint probability for x, y and o. We repeat this process until all scene elements are spatially arranged and select the hypothesis with the highest joint probability. (v) **Lowest-entropy (LE) - discrete**: Similarly, we perform beam search using the lowest entropy of the joint probability distribution for x, y and o.

4 Evaluation

4.1 Quantitative evaluation

We propose two measures to evaluate success in decoding 2D spatial arrangements from language. Both are similarity measures applied on normalized coordinates, hence the higher the better.

Absolute position similarity (abs. sim.) represents the average Euclidean distance between the ground truth and predicted position over all N clip-arts in the scene, defined as Gaussian function:

$$\mathcal{S}_{abs} = \frac{1}{N}\sum_{i=1}^{N} \exp\left(-\frac{\sqrt{(x_i^t - x_i^p)^2 + (y_i^t - y_i^p)^2}}{2\sigma}\right) \quad (7)$$

where $[x_i^t, y_i^t]$ and $[x_i^p, y_i^p]$ represent the ground truth and predicted position respectively, and σ is set to 0.2 as per Tan et al. (2018). We compute the similarity for both the original ground truth positions and the ground truth positions mirrored across the y axis, and subsequently take the maximum as the absolute similarity for that scene.

Relative position similarity (rel. sim.) focuses on the relative positioning of the objects with respect to each other. We compute two separate square matrices for the ground truth and predicted positions, M^t and M^p respectively, where the $M_{i,j}$

element is the Euclidean distance between the i-th and j-th object. Then, the similarity is the mean absolute difference between M^t and M^p, defined as a Gaussian function:

$$\mathcal{S}_{rel} = \frac{1}{(N-1)^2} \sum_{i=1}^{N} \sum_{j=1}^{N} \exp\left(-\frac{\left|M^t_{i,j} - M^p_{i,j}\right|}{2\sigma}\right) \tag{8}$$

where σ is again fixed to 0.2. Due to ambiguity (some object groups are vertically symmetrical) we evaluate the orientation o accuracy only within the scene completion setup. We further test the statistical significance between the average similarities of two methods with Welch's t-test ($p < 0.01$) (Welch, 1947) on the per scene similarities.

Baselines - we create two recurrent neural networks with attention baselines (Bahdanau et al., 2014). The first baseline (ATTN) uses an attention decoder, while the second propagates contextual information regarding the spatial positions (ATTN+RNN) (see appendix A). In both baselines, the clip-arts are ordered according to HO.

We measure the model's performance in terms of the similarity metrics in five different scenarios.

Full test set - we measure the performance of the discrete and continuous model on the full test set of 1002 semantically unique scenes. Table 1 reports results for all decoding strategies, when the model is and is not conditioned on the language (no-lang). For the no-lang model, we remove the language and use the concatenated `[CLS]` and `[SEP]` token indices to indicate that the language is excluded.

For all metrics, we see that the models which use a fine-grained decoding (HC, LE and HO) outperform the raw ones (SS, RO). We conclude that decoding order matters, and an orderless decoding (RO) is undesirable. Regardless of the decoding strategy, we observe significant gains when the model is conditioned on the language. This implies that our model is not trivially relying on dataset bias (e.g., mike usually wears a blue cap) when decoding the spatial arrangements (RQ 2). Furthermore, we observe a significant increase in both abs. sim. and rel. sim. with SR-BERT with HO decoding compared to its RNN counterpart – ATTN+RNN, in both the discrete and continuous model. This indicates the superiority of jointly attending on both the "future" and the "past" clip-arts, especially notable in the discrete model. Moreover, the continuous model outperforms the discrete model and is less sensitive to the decoding strategy – which we claim is due to the continuous model directly optimizing the evaluation metrics.

Scene completion (SC) - we formulate an inference scenario where we decode spatial arrangements for each group (explained in section 2) of clip-arts separately, conditioned on both the language and the remaining clip-arts. This is interpreted as a scene completion setting, e.g., what is the position and orientation of Mike and Jenny in the 2D space w.r.t. the other clip-arts if: "mike is holding a football", "mike wants to play football with jenny" and "jenny fell off the swing".

In Table 2 we see the highest abs. sim. and rel. sim. when we generate the arrangements for the "clothing" category conditioned on the other groups. On the contrary the lowest reported abs. sim and rel. sim are for the "animals" category. Finally, we see almost random o accuracy for the symmetrical object categories, with a major improvement for the object categories where their orientation matters.

Explicit vs. implicit relationships - we split the test set in 4 different subsets, where each contains scenes with $[0\%, 25\%]$, $(25\%, 50\%]$, $(50\%, 75\%]$ and $(75\%, 100\%]$ ratio of sentences that consist of explicit relations exclusively.

Figure 3 shows results with the discrete model with HC decoding and the continuous model using HO decoding. We observe that both the continuous and discrete model obtain steadily similar absolute and relative similarities as the ratio of explicit relations increases. This shows the robustness of our method in successfully coping with both explicit and implicit spatial language (RQ 3).

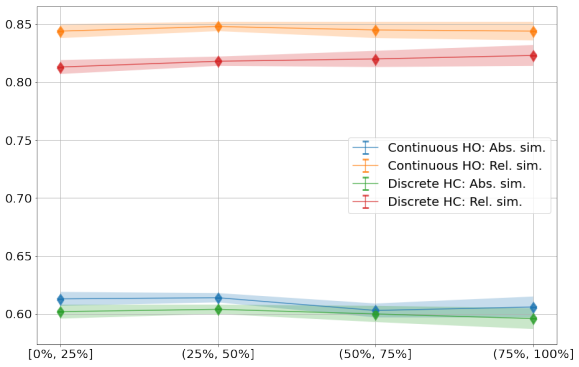

Figure 3: Reported metrics on test set splits according to the ratio of explicit relations.

Compositional generalization - to gain insight on the generalization ability, we split the test set by considering scenes that contain at least 2 sentences with a subject-relationship-object (S-R-O)

	Discrete		Continuous	
Method	Abs. sim.	Rel. sim.	Abs. sim.	Rel. sim.
SS	0.565 ± 0.002	0.769 ± 0.003	0.589 ± 0.002	0.814 ± 0.003
SS; no-lang	0.499 ± 0.002	0.695 ± 0.003	0.530 ± 0.002	0.779 ± 0.002
RO	0.585 ± 0.003	0.818 ± 0.003	0.603 ± 0.003	0.838 ± 0.003
RO; no-lang	0.523 ± 0.003	0.746 ± 0.003	0.546 ± 0.003	0.792 ± 0.002
HO	0.594 ± 0.003	0.823 ± 0.003	$\mathbf{0.611 \pm 0.003}$	$\mathbf{0.846 \pm 0.003}$
HO; no-lang	0.539 ± 0.003	0.745 ± 0.003	0.562 ± 0.003	0.794 ± 0.003
HC	$\mathbf{0.598 \pm 0.003}$	$\mathbf{0.826 \pm 0.003}$	—	—
HC; no-lang	0.534 ± 0.003	0.750 ± 0.003	—	—
LE	0.592 ± 0.003	0.825 ± 0.003	—	—
LE; no-lang	0.536 ± 0.003	0.751 ± 0.003	—	—
ATTN	0.565 ± 0.002	0.746 ± 0.002	0.579 ± 0.002	0.812 ± 0.002
ATTN+RNN	0.567 ± 0.002	0.746 ± 0.003	0.581 ± 0.002	0.813 ± 0.002

Table 1: Results of our models and the RNN baselines on the full test set.

Obj. group	Abs. sim.	Rel. sim.	*o* accuracy
Sky	0.655 ± 0.011	0.782 ± 0.010	48.4 ± 1.54
Large	0.671 ± 0.011	0.781 ± 0.012	57.8 ± 1.60
People	0.796 ± 0.006	0.857 ± 0.005	$\mathbf{86.7 \pm 0.97}$
Animals	0.661 ± 0.016	0.783 ± 0.017	70.5 ± 2.02
Clothing	$\mathbf{0.853 \pm 0.021}$	$\mathbf{0.900 \pm 0.021}$	71.2 ± 2.13
Food	0.772 ± 0.021	0.852 ± 0.023	46.9 ± 1.92
Toys	0.664 ± 0.015	0.786 ± 0.017	51.0 ± 1.80

Table 2: Per-object group results in the scene completion setup using the discrete model.

Method	Abs. sim.	Rel. sim.
D; HC; raw	$\mathbf{0.599 \pm 0.005}$	$\mathbf{0.817 \pm 0.005}$
D; HC; oo-spl	0.575 ± 0.005	0.799 ± 0.005
D; HC; in-spl	0.586 ± 0.005	0.807 ± 0.005
D; HC; no-lang	0.505 ± 0.005	0.724 ± 0.004
D; HC; filtered	0.581 ± 0.005	0.803 ± 0.005
C; HO; raw	$\mathbf{0.608 \pm 0.005}$	$\mathbf{0.844 \pm 0.004}$
C; HO; oo-spl	0.582 ± 0.005	0.817 ± 0.004
C; HO; in-spl	0.593 ± 0.005	0.828 ± 0.004
C; HO; no-lang	0.563 ± 0.004	0.795 ± 0.004
C; HO; filtered	0.586 ± 0.005	0.813 ± 0.004

Table 3: Results of the discrete (D) and continuous (C) model on the 5 distinct subsets defined for the compositional generalization experiments.

combination[5] that the model has not encountered during training. This yields 354 scenes from which we create 5 subsets:

(i) The **raw** scenes in their original form, which contain sentences with (S-R-O) combinations that the model has encountered, and at least two combinations that it has *not* encountered during training. (ii) From each of the raw scenes, we discard sentences consisting of (S-R-O) combinations that the model has encountered during training, while preserving the unseen ones. Hence, we are left with at least 2 sentences per scene, which are **out-of-sample (oo-spl)**. (iii) **In-sample (in-spl)** is the complementary of oo-spl and contains exclusively sentences consisting of (S-R-O) combinations encountered at training time while leaving out the out-of-sample sentences. (iv) The scenes without the language relations – **no-lang**. (v) We train a new model on a **filtered** training set where we leave out all sentences that have (S-R-O) combinations that appear in the raw scenes.

In Table 3 we observe that despite the effective similarity between the no-lang and oo-spl setting (in both the model is exposed to unfamiliar / no language), the difference in performance in favor of oo-spl is relatively big – especially with the discrete model. We also observe that the performance in the oo-spl setting degrades only moderately compared to the raw setting, and the oo-spl differs non-significantly from the in-spl setting. When comparing the raw and oo-spl, we must take into account that the model uses only partial scene descriptions in the oo-spl setting due to the held out sentences, which explains the moderate drop in performance. Finally, we observe that the filtered setting fares remarkably close to the raw setting, even though the model encounters all relations as out-of-sample. This observation suggests potentially great value in real-life scenarios where one is frequently exposed to unfamiliar spatial relations (RQ 2).

Complete scene generation pipeline - we extend our method to generate complete scenes, i.e.,

[5]For each sentence there is a S-R-O triplet in the dataset.

predicting the clip-arts and their spatial arrangements. We adopt a two-step approach where (1) we fine-tune BERT$_{\text{BASE}}$ as a backbone clip-art predictor, such that, given a language description of the scene x, the model outputs a vector of probabilities for each clip-art $\hat{y}$: $\hat{y} = \sigma(\text{Linear}(\text{BERT}(x)))$, and (2) we use SR-BERT to arrange the predicted clip-arts w.r.t. the language spatial relations. The linear layer in the clip-art predictor projects the text embedding from BERT's hidden space to 126 (number of unique clip-arts), and σ is the sigmoid non-linearity, thresholded at 0.35 during inference. We compute the per-object precision (Prec) and recall (Rec), classification accuracies for poses (Pose) and expressions (Expr), and abs. sim. for the object positions[6]. When generating the scene arrangements, we first provide the predicted clip-arts to SR-BERT and then arrange them on the scene. We then find the common clip-arts between the predictions and the ground truths and measure abs. sim., as per Tan et al. (2018)[7]. We train a new discrete SR-BERT model on Tan et al. (2018)'s data splits and perform inference using HC decoding.

Method	Prec	Rec	Pose	Expr	Abs. sim.
(Zitnick et al., 2013)	72.2	65.5	40.7	30.0	0.449
(Tan et al., 2018)	76.0	69.8	**41.8**	37.5	0.409
ClipPredict + SR-BERT	**82.7**	**72.5**	40.4	**38.0**	**0.512**

Table 4: Per-object precision and recall, pose and expression classification accuracies, and abs. sim. using the test split provided by Tan et al. (2018).

In Table 4 we observe that our pipeline outperforms the concurrent methods of Tan et al. (2018); Zitnick et al. (2013) in terms of precision, recall and expression accuracy, while it falls slightly short in terms of pose accuracy. Moreover, SR-BERT outperforms the concurrent methods according to abs. sim. by a large margin, which measures spatial reasoning. We want to stress however, that because of the simplicity of SR-BERT, it can be trivially plugged in within a more powerful abstract scene generation from a language pipeline.

4.2 Qualitative evaluation

In Figure 4 we compare the spatial arrangements between the scene elements with or without the language relations. E.g., if we exclude the language, both the middle-left and the bottom-left are plausible scenes. However, when the language is present, "jenny", the "football" and "mike" take upon certain positions / orientations to satisfy the imposed language relations. Figure 4 (right) demonstrates the scene completion feature of our method. We see that the most probable location of the "owl" is in the tree, which is intuitive. However, when conditioning on the sentences "mike is afraid of the owl" and "mike is trying to play with the owl", the distribution shifts to the "owl" being in the sandbox. This indicates that the model gains understanding of how implicit spatial relationships transfer in the 2D spatial domain (RQ 3).

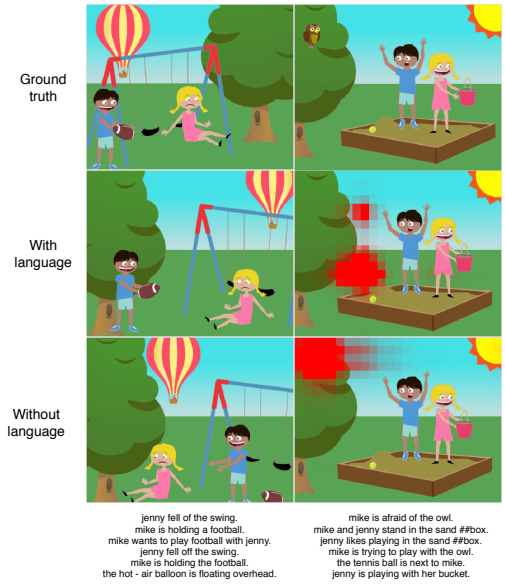

Figure 4: Ground truth (top), generated scenes (middle-left, bottom-left) and heat-maps (middle-right, bottom-right) with and without conditioning on the language.

We also qualitatively evaluate the complete scene generation pipeline on the data split provided by Tan et al. (2018). In Figure 5 we observe scenes generated in a two step process by (1) predicting the clip-arts, and (2) arranging them using SR-BERT. In both scenarios we observe predicted clip-arts which are relatively inline with scene descriptions. Furthermore, despite providing predictions which do not perfectly resemble the ground truths, the scene arrangements generated with SR-BERT obtain consistent quality w.r.t. the cases when the ground truth clip-arts are provided as input.

4.3 User study

We conduct a user study to evaluate to what extent the generated spatial arrangements align with hu-

[6]The `U-obj` coord metric of (Tan et al., 2018) is equivalent to our absolute position similarity.

[7]Contrary to the other experiments, here we measure abs. sim. only w.r.t. the default ground truth positions and not the mirrored ones for a pessimistic comparison.

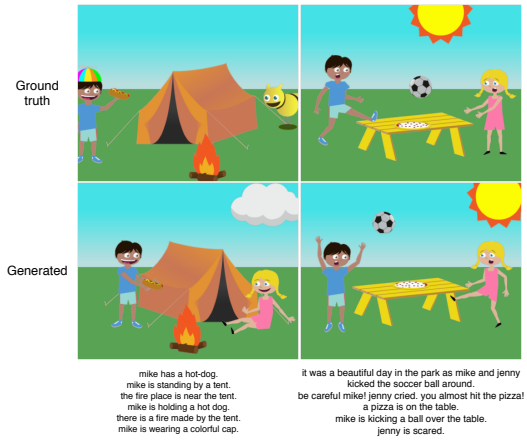

Figure 5: Ground truth (top) and generated scenes (bottom) by (1) predicting the clip-arts and (2) using SR-BERT to arrange them on the 2D canvas.

man judgement (RQ 1). We randomly select 100 samples (∼10% of the full test set) and generate the spatial arrangements using the continuous model with RO and HO and the discrete model with RO and HC decoding. The participants are presented with a scene together with the corresponding sentences that imply spatial relatedness between the clip-arts[8], and are asked to select the sentences which are spatially true for the scene. We report the macro-average per scene results for each model plus the ground truth in Figure 6.

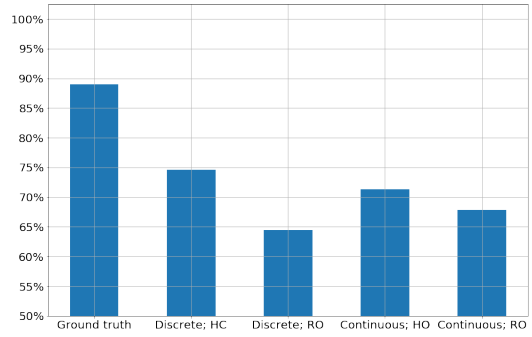

Figure 6: Per-scene macro-average of the accepted sentences from the participants in the user study.

Irrespective of the model and decoding strategy, all models perform well compared to the ground truth, and users found that at least 64% of the sentences are spatially true for the generated scenes while the ground truth scenes are at the 88% mark, with a fair agreement (k=0.29) between raters as per Fleiss' Kappa score (Fleiss and Cohen, 1973).

[8]To avoid an overly optimistic estimate, we remove sentences which are always correct, i.e., sentences that do not imply any spatial connection between the clip-arts e.g., "mike is smiling".

Despite the non-significant difference between the RO and HO decoding with the continuous model and RO and HC with the discrete model in Table 1, the continuous model with HO decoding and discrete model with HC decoding outperform the continuous and discrete model with RO decoding respectively by a large margin. We see a less amplified difference of 4% between the HO and RO decoding with the continuous model compared to the discrete model which reports 10% difference between the HC and RO decoding. This is due to the continuous model being less sensitive to the decoding order. This strengthens our hypothesis that the decoding order is important, which comes automatically with the discrete model and HC decoding, while it needs to be injected manually in the continuous model with HO decoding. Consequently, the best performance is achieved with the discrete model with HC decoding – above 74.6% accepted spatial relations and the continuous model with HO decoding – 71.3% accepted spatial relations.

5 Related work

Spatial understanding in vision is mainly limited to scenes that have strong priors on the relative object's locations, e.g., indoor home environments. Choi et al. (2013)'s model reasons about object interactions in 3D and performs object detection, layout estimation and scene classification, limited to images of indoor home environments. Xu et al. (2013)'s method maps indoor scene sketches in 3D space. Similar to our approach, they do not limit the number of objects. Kermani et al. (2016) synthesize 3D indoor home environments from RGB-D. Fisher et al. (2012) also reason about spatial arrangements using 3D indoor home environments by building a probabilistic model and then sample diverse 3D scenes. The work of Jiang et al. (2012) explores human-object relationships in 3D indoor scenes. Given a scene, Zhao et al. (2016) synthesize new scenes that preserve the nature of the original spatial relations by replacing the objects. On the other hand, our work is not limited in terms of the relationships type, i.e., explores human-object, object-object and human-human relationships in clip-arts scenes.

Spatial understanding in language often suffers from processing structured language or again, using scenes that have strong priors on the relative object's locations. Chang et al. (2014) infer spatial relationships not explicitly stated in natural lan-

guage, and generate 3D indoor home environments. Contrary to our work, the 3D indoor home environments are limited in terms of implicit spatial language (e.g., wearing, holding, etc.). Chang et al. (2017) decode language spatial relations to 3D indoor home environment layouts, by firstly selecting the objects and then arranging them. We, however, loosen the object selection part, and provide the objects and the language spatial relations as input. In contrast with our work, they also consider only object-object explicit relations. Collell et al. (2018) introduced the notion of implicit spatial relations, expanding on prior research limited to explicit relations, yet they restrict to two objects and a single relationship in a structured format. Although not related to spatial understanding of scenes, Dan et al. (2020b) create a spatial representation language to describe spatial configurations, while Kordjamshidi et al. (2010, 2011) tackle spatial role labeling with a relational learning framework.

Multimodal spatial understanding mostly consists of works that employ a spatial reasoning module in their pipeline, yet proper spatial understanding is not their main goal but rather a secondary sub-problem, hence less emphasis is placed on spatial correctness and evaluation. E.g., Johnson et al. (2018); Herzig et al. (2019) generate images from text descriptions with an intermediate scene layout generation step. Moreover, Lee et al. (2019); Li et al. (2019a,c); Hong et al. (2018) explicitly focus on generating high quality multi-object and multi-relationship 2D scene layouts from natural language, without limiting the type or number of relationships. However, their layout module does not aim for a precise depiction of the scene arrangements, but rather provides a rough outline for the subsequently generated image. The closest works to ours are Tan et al. (2018); Zitnick et al. (2013) who use the same dataset of Zitnick and Parikh (2013) and generate realistically-looking scenes, given a language description of the scene's spatial arrangements. Despite showcasing that our method is superior to theirs, it can also complete partial scenes and is more extensively evaluated on the Abstract Scenes dataset. Dan et al. (2020a) predict the relationship word given the image, a bounding box, and the subject and object words by using a spatial model to filter the predictions of a fine-tuned BERT model. Their model does not decode language to 2D spatial arrangements while reasoning about their position. Finally, Ghanimifard and Dobnik (2019) generate spatial image descriptions to investigate what kind of spatial bottom-up knowledge, benefits the top-down methods the most.

6 Conclusion

In this paper, we address the problem of spatial understanding by predicting spatial arrangements of scenes given their natural language descriptions. This work advances towards general spatial understanding of visual scenes and language by addressing the limitations of prior work: (i) modelling only pairwise relationships; (ii) using scene environments with strong priors on (relative) object placements (e.g., indoor home environments); (iii) use of structured language (instead of natural language). We proposed a novel architecture – SR-BERT, for which we empirically demonstrate that it is capable of reasoning about an arbitrary number of objects and the relationships between them in the 2D space. SR-BERT's spatial reasoning is irrespective of the spatial language type employed (explicit or implicit) and effectively generalizes to out-of-sample instances – a frequently occurring real-world situation where one encounters novel (spatial) scene descriptions.

The first limitation of our approach is that it is restricted to 2D spatial reasoning despite computer graphics data being prevalently 3D. However, we kept a simple setting for our first study of spatial reasoning while our approach can trivially be extended to work with 3D data. Second, we assume a fixed number of 126 clip-art categories and canvas of 500x400, which is a rather idealized setting considering that actual 2D/3D modelling is performed with: (1) larger number of categories, and (2) bigger canvases. Finally, scene layout generation is an ill-posed problem, i.e., there are multiple valid layouts conditioned on a scene description. Therefore, an ideal approach should not generate a deterministic layout, but rather address the uncertainty in both the model's output and the evaluation.

Acknowledgments

We acknowledge funding from the Flemish Government under the Onderzoeksprogramma Artificiële Intelligentie (AI) Vlaanderen programme. This work has also been supported by the CHIST-ERA EU project MUSTER[9].

[9] www.chistera.eu/projects/muster

References

Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. 2016. Layer normalization. *arXiv preprint arXiv:1607.06450*.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*.

Angel Chang, Manolis Savva, and Christopher D Manning. 2014. Learning spatial knowledge for text to 3d scene generation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2028–2038.

Angel X Chang, Mihail Eric, Manolis Savva, and Christopher D Manning. 2017. Sceneseer: 3d scene design with natural language. *arXiv preprint arXiv:1703.00050*.

Yen-Chun Chen, Linjie Li, Licheng Yu, Ahmed El Kholy, Faisal Ahmed, Zhe Gan, Yu Cheng, and Jingjing Liu. 2019. Uniter: Learning universal image-text representations. *arXiv preprint arXiv:1909.11740*.

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*.

Wongun Choi, Yu-Wei Chao, Caroline Pantofaru, and Silvio Savarese. 2013. Understanding indoor scenes using 3d geometric phrases. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 33–40.

Guillem Collell, Luc Van Gool, and Marie-Francine Moens. 2018. Acquiring common sense spatial knowledge through implicit spatial templates. In *Thirty-Second AAAI Conference on Artificial Intelligence*.

Soham Dan, Hangfeng He, and Dan Roth. 2020a. Understanding spatial relations through multiple modalities. In *Proceedings of The 12th Language Resources and Evaluation Conference*, pages 2368–2372.

Soham Dan, Parisa Kordjamshidi, Julia Bonn, Archna Bhatia, Zheng Cai, Martha Palmer, and Dan Roth. 2020b. From spatial relations to spatial configurations. In *Proceedings of The 12th Language Resources and Evaluation Conference*, pages 5855–5864.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Matthew Fisher, Daniel Ritchie, Manolis Savva, Thomas Funkhouser, and Pat Hanrahan. 2012. Example-based synthesis of 3d object arrangements. *ACM Transactions on Graphics (TOG)*, 31(6):1–11.

Joseph L Fleiss and Jacob Cohen. 1973. The equivalence of weighted kappa and the intraclass correlation coefficient as measures of reliability. *Educational and psychological measurement*, 33(3):613–619.

Mehdi Ghanimifard and Simon Dobnik. 2019. What goes into a word: generating image descriptions with top-down spatial knowledge. In *Proceedings of the 12th International Conference on Natural Language Generation*, pages 540–551, Tokyo, Japan. Association for Computational Linguistics.

Marjan Ghazvininejad, Omer Levy, Yinhan Liu, and Luke Zettlemoyer. 2019a. Constant-time machine translation with conditional masked language models. *arXiv preprint arXiv:1904.09324*.

Marjan Ghazvininejad, Omer Levy, Yinhan Liu, and Luke Zettlemoyer. 2019b. Mask-predict: Parallel decoding of conditional masked language models. 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 6114–6123.

Dan Hendrycks and Kevin Gimpel. 2016. Gaussian error linear units (gelus). *arXiv preprint arXiv:1606.08415*.

Roei Herzig, Amir Bar, Huijuan Xu, Gal Chechik, Trevor Darrell, and Amir Globerson. 2019. Learning canonical representations for scene graph to image generation. *arXiv preprint arXiv:1912.07414*.

Seunghoon Hong, Dingdong Yang, Jongwook Choi, and Honglak Lee. 2018. Inferring semantic layout for hierarchical text-to-image synthesis. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7986–7994.

Yun Jiang, Marcus Lim, and Ashutosh Saxena. 2012. Learning object arrangements in 3d scenes using human context. *arXiv preprint arXiv:1206.6462*.

Justin Johnson, Agrim Gupta, and Li Fei-Fei. 2018. Image generation from scene graphs. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1219–1228.

Jungo Kasai, James Cross, Marjan Ghazvininejad, and Jiatao Gu. 2020. Parallel machine translation with disentangled context transformer. *arXiv preprint arXiv:2001.05136*.

Z Sadeghipour Kermani, Zicheng Liao, Ping Tan, and H Zhang. 2016. Learning 3d scene synthesis from annotated rgb-d images. In *Computer Graphics Forum*, volume 35, pages 197–206. Wiley Online Library.

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

Parisa Kordjamshidi, Marie-Francine Moens, and Martijn van Otterlo. 2010. Spatial role labeling: Task definition and annotation scheme. In *Proceedings of the Seventh conference on International Language Resources and Evaluation (LREC'10)*, pages 413–420. European Language Resources Association (ELRA).

Parisa Kordjamshidi, Martijn Van Otterlo, and Marie-Francine Moens. 2011. Spatial role labeling: Towards extraction of spatial relations from natural language. *ACM Transactions on Speech and Language Processing (TSLP)*, 8(3):1–36.

Carolin Lawrence, Bhushan Kotnis, and Mathias Niepert. 2019. Attending to future tokens for bidirectional sequence generation. *arXiv preprint arXiv:1908.05915*.

Hsin-Ying Lee, Weilong Yang, Lu Jiang, Madison Le, Irfan Essa, Haifeng Gong, and Ming-Hsuan Yang. 2019. Neural design network: Graphic layout generation with constraints. *arXiv preprint arXiv:1912.09421*.

Jason Lee, Elman Mansimov, and Kyunghyun Cho. 2018. Deterministic non-autoregressive neural sequence modeling by iterative refinement. *arXiv preprint arXiv:1802.06901*.

Boren Li, Boyu Zhuang, Mingyang Li, and Jian Gu. 2019a. Seq-sg2sl: Inferring semantic layout from scene graph through sequence to sequence learning. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 7435–7443.

Liunian Harold Li, Mark Yatskar, Da Yin, Cho-Jui Hsieh, and Kai-Wei Chang. 2019b. Visualbert: A simple and performant baseline for vision and language. *arXiv preprint arXiv:1908.03557*.

Wenbo Li, Pengchuan Zhang, Lei Zhang, Qiuyuan Huang, Xiaodong He, Siwei Lyu, and Jianfeng Gao. 2019c. Object-driven text-to-image synthesis via adversarial training. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 12174–12182.

Ilya Loshchilov and Frank Hutter. 2017. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*.

Jiasen Lu, Dhruv Batra, Devi Parikh, and Stefan Lee. 2019. Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. In *Advances in Neural Information Processing Systems*, pages 13–23.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. 2019. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems*, pages 8024–8035.

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15(1):1929–1958.

Weijie Su, Xizhou Zhu, Yue Cao, Bin Li, Lewei Lu, Furu Wei, and Jifeng Dai. 2019. Vl-bert: Pre-training of generic visual-linguistic representations. *arXiv preprint arXiv:1908.08530*.

Chen Sun, Austin Myers, Carl Vondrick, Kevin Murphy, and Cordelia Schmid. 2019. Videobert: A joint model for video and language representation learning. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 7464–7473.

Fuwen Tan, Song Feng, and Vicente Ordonez. 2018. Text2scene: Generating compositional scenes from textual descriptions. *arXiv preprint arXiv:1809.01110*.

Hao Tan and Mohit Bansal. 2019. Lxmert: Learning cross-modality encoder representations from transformers. *arXiv preprint arXiv:1908.07490*.

Iulia Turc, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Well-read students learn better: On the importance of pre-training compact models. *arXiv preprint arXiv:1908.08962*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

Alex Wang and Kyunghyun Cho. 2019. Bert has a mouth, and it must speak: Bert as a markov random field language model. *arXiv preprint arXiv:1902.04094*.

Bernard L Welch. 1947. The generalization ofstudent's' problem when several different population variances are involved. *Biometrika*, 34(1/2):28–35.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, et al. 2019. Transformers: State-of-the-art natural language processing. *arXiv preprint arXiv:1910.03771*.

Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, et al. 2016. Google's neural machine translation system: Bridging the gap between human and machine translation. *arXiv preprint arXiv:1609.08144*.

Kun Xu, Kang Chen, Hongbo Fu, Wei-Lun Sun, and Shi-Min Hu. 2013. Sketch2scene: sketch-based co-retrieval and co-placement of 3d models. *ACM Transactions on Graphics (TOG)*, 32(4):1–15.

Xi Zhao, Ruizhen Hu, Paul Guerrero, Niloy Mitra, and Taku Komura. 2016. Relationship templates for creating scene variations. *ACM Transactions on Graphics (TOG)*, 35(6):1–13.

Yukun Zhu, Ryan Kiros, Rich Zemel, Ruslan Salakhutdinov, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. 2015. Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. In *Proceedings of the IEEE international conference on computer vision*, pages 19–27.

C Lawrence Zitnick and Devi Parikh. 2013. Bringing semantics into focus using visual abstraction. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3009–3016.

C Lawrence Zitnick, Devi Parikh, and Lucy Vanderwende. 2013. Learning the visual interpretation of sentences. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1681–1688.