A Recurrent neural network baselines

The recurrent neural network baselines consists of two bidirectional GRU recurrent neural networks (Cho et al., 2014), each with a hidden size of 256, where one is used for obtaining a contextual embeddings of the text and the other for the clip-arts. Prior to the RNN, the clip-arts are ordered according to the HO, and an embedding of size 256 is obtained for every clip-art and word in the sentences. In both models, an attention module attends on every sentence word with respect to every clip-art in a sequential manner when generating the spatial arrangements of that clip-art. The ATTN+RNN baseline, uses an extra GRU recurrent neural network for propagating contextual information of the generated spatial arrangements. The hidden size of the attention module is kept the same as in the GRU recurrent neural network – 256. Both models are trained for 300 epochs with a fixed learning rate of $2e-5$, while the model with the best performance on the validation set is used for inference on the test set. There are no other forms of regularization. As in SR-BERT, we implement two variants of the ATTN and ATTN+RNN models, namely a continuous and a discrete version.

B Data augmentation

Due to the limited data we have at hand, we are employing several data augmentation strategies to artificially increase the quantity of data, while preserving the meaning contained within a single scene. Furthermore, we want to impose a greater importance on the relative positioning of the elements, as well as obtain a model which is scene mirroring invariant and invariant of the order of the language spatial relationships. To this end, the data augmentations we use are:

- In the dataset we use, each scene is paired with a set of ~ 6 sentences, where each sentence is entirely self-contained ("The cat is on the bench"), which implies that the order of the sentences does not matter. Therefore, we randomly shuffle the sentences before concatenating and tokenizing them.

- For each valid scene in the dataset, the mirrored scene according to the y axis is also a valid scene. Therefore, we randomly mirror each scene with 50% probability by reverting the x coordinates - $|width - x|$, and reverting the orientation - $|1 - o|$ of each object.

- With 50% probability we move all elements in the scene up, down, left or right, by a random number of quantization intervals, with 25% probability for each movement. The number of quantization intervals is sampled uniformly from $[0, max_x)]$ for x and $[0, max_y]$ for y.

C Experimental setup

The backbone of all our models is the $\text{BERT}_{\text{BASE}}$ variant from Devlin et al. (2018), pre-trained on the BooksCorpus (Zhu et al., 2015) and English Wikipedia. We train all SR-BERT models using MPM as a training objective for 300 epochs, while we train the clip-arts predictor model for 50 epochs. We use is Adam (Kingma and Ba, 2014) with a learning rate of $2e-5$ for training the SR-BERT models, and AdamW (Loshchilov and Hutter, 2017) with a learning rate of $5e-5$ and weight decay of $1e-2$ for the clip-art predictor model since it sped up convergence significantly. For all SR-BERT models, we empirically set the scaling factor λ to $\frac{1}{3}$ to obtain the average of the three spatial embeddings. Apart from applying early stopping, i.e., saving the model with the best performance on the validation set, we do not tune our hyperparameters. For the data augmentation max_x is equal to 25, and max_y to 20, since we use a fixed quantization interval of 20 on a scene size of 500×400. We use a separate decoding strategy during model selection on the validation set, so that we can make an unbiased estimate about the performance of the aforementioned decoding strategies. Namely, when performing inference on the validation set, we generate the spatial arrangement of each scene element by conditioning it on the ground truth spatial positions of all other scene elements in addition to the sentences. The deep learning library of choice is PyTorch (Paszke et al., 2019) alongside the HuggingFace Transformers package (Wolf et al., 2019).

D Additional quantitative evaluation

From the full dataset, we randomly sample 1002 scenes for testing, 1002 scenes for model selection and we use the remaining 7989 scenes for training the models. We report the results in Table 5. In addition, we include result of three naive baselines: (i) Random - we generate a set of random $[x, y]$ coordinates by uniformly sampling from $[0, 500]$ for x and $[0, 400]$ for y. (ii) Center - we place every object in the scene center $[250, 200]$. (iii) Train-set average - for each object category we compute the average across the x and y axes on the training set. Then, during inference, we impute those values as the predicted positions.

Additionally, we train a new discrete SR-BERT model on the dataset splits provided by Tan et al. (2018) and we report the absolute and relative position similarities on the test set. We follow the original evaluation methodology, i.e., 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.

We also investigate the effect of the model size by performing inference with $\text{BERT}_{\text{MEDIUM}}$, $\text{BERT}_{\text{SMALL}}$ and $\text{BERT}_{\text{MINI}}$ provided by Turc et al. (2019). Namely, for each of the three model sizes, we train a discrete and a continuous model and perform inference with HC and HO decoding on the full test set and report the results in Table 7.

Finally, we investigate how the masking percentage, initially fixed to 15% for all models in the main paper, affects the models' performance. We report results in Table 8 with continuous and discrete model based on $\text{BERT}_{\text{BASE}}$ by increasing the masking percentage during training to 30%.[10]

E Additional qualitative evaluation

We select 10 random samples from the test and generate the spatial arrangements given the language using our two best models – discrete with HC decoding and continuous with HO decoding. We report the results in Figure 8.

[10] Since the discrete model trained with a masking percentage of 30% and the discrete model of Medium size differ not-significantly from the best discrete model in the main paper, we make all experiments in the main paper with the $\text{BERT}_{\text{BASE}}$ discrete model trained with a masking percentage of 15%.

F Additional information about the user study

We conduct the user study on Amazon Mechanical Turk where for each assignment (generated scene) there are 3 distinct participants voting for the spatial correctness of the sentences. One such assignment can be seen on Figure 7. We collect the results from the user study such that in case a sentence has been selected as *True* by at least two participants, we deem that sentence as *True* for the scene, and *False* otherwise. Then, for each scene we compute the average number of accepted spatial arrangements as the fraction of accepted sentences for that scene. Finally, we obtain the macro-average over the whole subset of scenes.

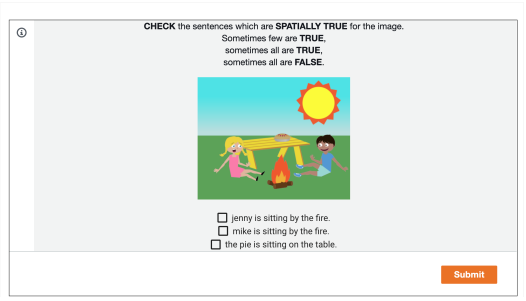

Figure 7: Sample task from the user study we conduct on Amazon Mechanical Turk.

	Discrete		Continuous	
Method	Abs. sim.	Rel. sim.	Abs. sim.	Rel. sim.
SS	0.566 ± 0.002	0.756 ± 0.003	0.563 ± 0.002	0.786 ± 0.002
SS; no-lang	0.439 ± 0.002	0.623 ± 0.003	0.526 ± 0.002	0.767 ± 0.002
RO	0.591 ± 0.003	0.813 ± 0.003	0.576 ± 0.003	0.814 ± 0.003
RO; no-lang	0.496 ± 0.002	0.706 ± 0.003	0.526 ± 0.003	0.790 ± 0.002
HO	0.597 ± 0.003	0.821 ± 0.003	$\mathbf{0.589 \pm 0.003}$	$\mathbf{0.816 \pm 0.003}$
HO; no-lang	0.495 ± 0.002	0.703 ± 0.003	0.555 ± 0.002	0.792 ± 0.002
HC	$\mathbf{0.605 \pm 0.003}$	0.825 ± 0.003	—	—
HC; no-lang	0.501 ± 0.002	0.709 ± 0.003	—	—
LE	0.601 ± 0.003	$\mathbf{0.826 \pm 0.003}$	—	—
LE; no-lang	0.499 ± 0.002	0.711 ± 0.003	—	—
ATTN	0.572 ± 0.002	0.747 ± 0.003	0.579 ± 0.002	0.809 ± 0.002
ATTN+RNN	0.575 ± 0.003	0.751 ± 0.003	0.578 ± 0.002	0.810 ± 0.003
Random	0.384 ± 0.002	0.666 ± 0.002	0.384 ± 0.002	0.666 ± 0.002
Center	0.456 ± 0.002	0.412 ± 0.002	0.456 ± 0.002	0.412 ± 0.002
Train-set average	0.535 ± 0.002	0.687 ± 0.003	0.535 ± 0.002	0.687 ± 0.003

Table 5: Abs. sim. and rel. sim. on the full test set when the dataset split is done randomly.

	Discrete	
Method	Abs. sim.	Rel. sim.
SS	0.580 ± 0.002	0.777 ± 0.003
SS; no-lang	0.485 ± 0.002	0.666 ± 0.003
RO	0.607 ± 0.003	0.824 ± 0.003
RO; no-lang	0.526 ± 0.003	0.720 ± 0.003
HO	0.608 ± 0.003	0.826 ± 0.003
HO; no-lang	0.531 ± 0.003	0.715 ± 0.003
HC	$\mathbf{0.616 \pm 0.003}$	$\mathbf{0.835 \pm 0.003}$
HC; no-lang	0.532 ± 0.003	0.724 ± 0.003
LE	0.608 ± 0.003	0.829 ± 0.003
LE; no-lang	0.529 ± 0.003	0.718 ± 0.003

Table 6: Abs. sim. and rel. sim. on the test set provided by Tan et al. (2018).

	Discrete		Continuous	
Method	Abs. sim.	Rel. sim.	Abs. sim.	Rel. sim.
$\text{BERT}_{\text{BASE}}$; HC	0.598 ± 0.003	0.826 ± 0.003	—	—
$\text{BERT}_{\text{BASE}}$; HO	0.594 ± 0.003	0.826 ± 0.003	$\mathbf{0.611 \pm 0.003}$	$\mathbf{0.846 \pm 0.003}$
$\text{BERT}_{\text{MEDIUM}}$; HC	$\mathbf{0.614 \pm 0.003}$	$\mathbf{0.829 \pm 0.003}$	—	—
$\text{BERT}_{\text{MEDIUM}}$; HO	0.608 ± 0.003	0.824 ± 0.003	0.583 ± 0.003	0.824 ± 0.003
$\text{BERT}_{\text{SMALL}}$; HC	0.596 ± 0.003	0.808 ± 0.003	—	—
$\text{BERT}_{\text{SMALL}}$; HO	0.590 ± 0.003	0.804 ± 0.003	0.571 ± 0.002	0.808 ± 0.002
$\text{BERT}_{\text{MINI}}$; HC	0.593 ± 0.003	0.794 ± 0.003	—	—
$\text{BERT}_{\text{MINI}}$; HO	0.587 ± 0.003	0.79 ± 0.003	0.565 ± 0.002	0.787 ± 0.002

Table 7: Abs. sim. and rel. sim. on the full test set obtained with smaller BERT variants.

	Discrete		Continuous	
Method	Abs. sim.	Rel. sim.	Abs. sim.	Rel. sim.
15%; HC	0.598 ± 0.003	0.826 ± 0.003	—	—
15%; HO	0.594 ± 0.003	0.823 ± 0.003	0.611 ± 0.003	0.846 ± 0.003
30%; HC	$\mathbf{0.608 \pm 0.003}$	$\mathbf{0.829 \pm 0.003}$	—	—
30%; HO	0.599 ± 0.003	0.824 ± 0.003	$\mathbf{0.612 \pm 0.003}$	$\mathbf{0.847 \pm 0.003}$

Table 8: Abs. sim. and rel. sim. on the full test set with models trained with different masking percentages.

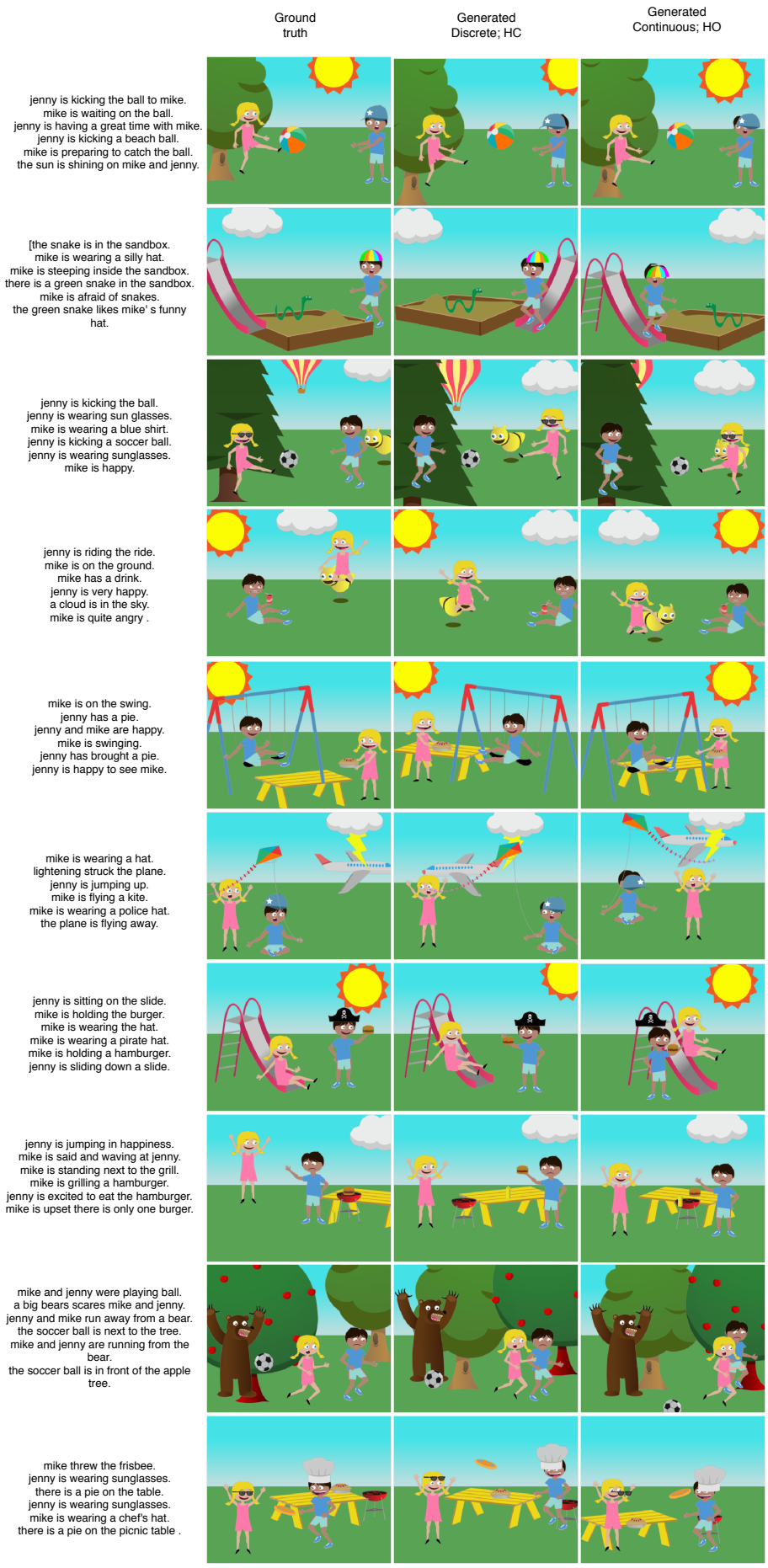

Figure 8: Generated spatial arrangements conditioned on language on 10 random samples from the test set.