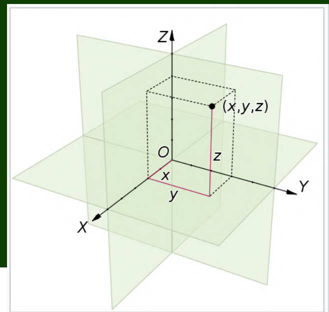
Table of Content

- Challenges and Motivating Applications
- Spatial Representations
- Spatial Reasoning
- Spatial Information Extraction
- Downstream Tasks
  - Visual Question Answering
  - Navigation and Instruction Following
  - Dialogue Systems
  - Talking to Self-driving Cars

- In what follows, focus on how spatial language could be understood in a way humans do
- Illustrated with neural network approaches that model distributed representations

Human Spatial Cognition: Realization



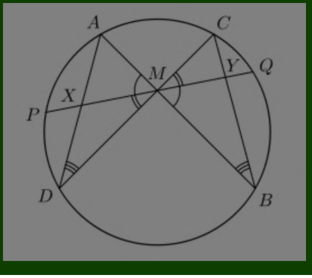
- Through grounding of language in visual perceptual world
- Through imagination of language in visual perceptual world
- Though reasoning in a geometric 2D or 3D space

=> Inspiration source for processing spatial language

- Motivated by many practical applications

Aflalo, T. N. & Graziano, M. S. A. (2008). Four-dimensional spatial reasoning in humans. *Journal of Experimental Psychology: Human Perception and Performance*, 34(5), 1066–1077.

Study of space



- In antiquity the study of space emerged among the ancient Babylonians and Greeks and led to Euclidean geometry
- The next breakthrough was probably the development of analytic geometry by René Descartes and the projective geometry by Girard Desargues in the 17th century
- In the 19th century non-Euclidean geometries were developed extending the concept of space beyond what could be intuited through everyday perception
- Today neuroscientist John O'Keefe contributed pioneering work on mammalian spatial cognition: three-dimensional Euclidean construction is inherent to the human nervous system
- The human experience of space includes knowledge relating to size, shape, location and distribution of entities in a 3D environment

Vyvyan Evans & Paul Anthony Chilton (2010). *Language, Cognition and Space: The State of the Art and New Directions*. Equinox.

Implicit versus explicit spatial language

- Focus on spatial understanding of language and representing language with **spatial templates** = regions of acceptability of two objects under a spatial relationship
- Prior work restricts spatial templates to language that **explicitly** uses spatial cues (e.g., “glass *on* table”)
- We extend this concept to **implicit** spatial language, i.e., those relationships (generally actions) for which the spatial arrangement of the objects is only implicitly implied (e.g., “man *riding* horse”) => requires significant commonsense spatial understanding

Logan, G. D., & Sadler, D. D. (1996). A computational analysis of the apprehension of spatial relations. In P. Bloom, M. A. Peterson, L. Nadel, & M. F. Garrett (Eds.), *Language, Speech, and Communication: Language and Space* (p. 493–529). The MIT Press.

Reinhard Moratz &Thora Tenbrink (2006). Spatial reference in linguistic human-robot interaction: Iterative, empirically supported development of a model of projective relations. *Spatial Cognition and Computation*, 6(1), 63–106.

Implicit versus explicit spatial language

waiting on the stairs

up on the right

Fig. 2. "About 20 kids in traditional clothing and hats waiting on stairs. A house and a green wall with gate in the background. A sign saying that plants can't be picked up on the right."

Implicit versus explicit spatial language

A girl rides a horse

- Where is the horse located, where is the girl located in relation to the horse?

- **Can we build suitable representations in the physical space that capture this knowledge and potentially make inferences with it?**

Implicit versus explicit spatial language

Depending on the context, spatial language might have different meaning in terms of targeted geometry

a man riding a motorcycle in front of an orange bus

The distance between the man and the motorcycle is usually much smaller in a city environment compared to a highway environment

Implicit spatial information - Dynamics

- **Spatio-temporal change** is encoded in verbs
- Pre-conditions of an action:
  - "Shut the door!" door is in open position
  - "Jan arrived in Prague." Jan is not in Prague
- Post-conditions of an action:
  - "Shut the door!" door is in closed position
  - "Jan arrived in Prague." Jan is now in Prague
- Physical consequences of actions

Not treated in this tutorial: but interesting research topics

Distributed representations

- Current neural network models create distributed representations
- Geoffrey Hinton, James L. McClelland and David Everett Rumelhart (1986): "Each entity is represented by a pattern of activity distributed over many computing elements, and each computing element is involved in representing many different entities."
    - Each concept is represented by many neurons
    - Each neuron participates in the representation of many concepts

  ⇔ Localist representations: one neuron or node is dedicated for each entity/thing

Distributed ⇔ Symbolic representations

Have the advantage:

- To be robust in processing tasks
- To be able to capture context
- Easier to scale up
- More useful for connecting to neuroscience
- Better for perceptual problems
- ...

Have the advantage:

- Easier to explain to humans
- Easier to code
- Better for abstract concepts
- Used in communication with humans
- ...

Visualizing language content

- It is well-known that humans "imagine" language content in a visual space
- It is well-known that humans reason in spatial visual space
- How to predict the spatial configurations and location of objects, actions, and their attributes in a 2D or 3D space?

= test of how well does the system understands spatial language

Visualizing language content

- This work has potential for real-time language understanding in a visual context:
  - Language communication to robots, machines, self-driving cars, …
  - Translation of spatial language to 2D or 3D space opens possibilities of fast **quantitative reasoning in such a space**, which can complement qualitative symbolic representations and reasoning
- This work is a step towards opening the black box of neural models applied to language processing by visualizing the interpreted content

Visualizing the location of an object

- We propose the task of:
  - Given a structured text input of the form (Subject, Relationship, Object) = (S,R,O)
  - Predict the 2D relative spatial arrangement of two objects (output)
- Train the task in a supervised setting:
  - Training set of image-text pairs, where the size and location of bounding boxes of objects in images serve as ground truth
- = a spatial “question-answering" task where the question consists in a spatial commonsense query such as *where is the* “man" *located with respect to a* “horse" *when a* “man" *is* “feeding" *the* “horse"?
- The answer is a 2D “imagined” representation in contrast with a sentence/word as typically done in question-answering tasks

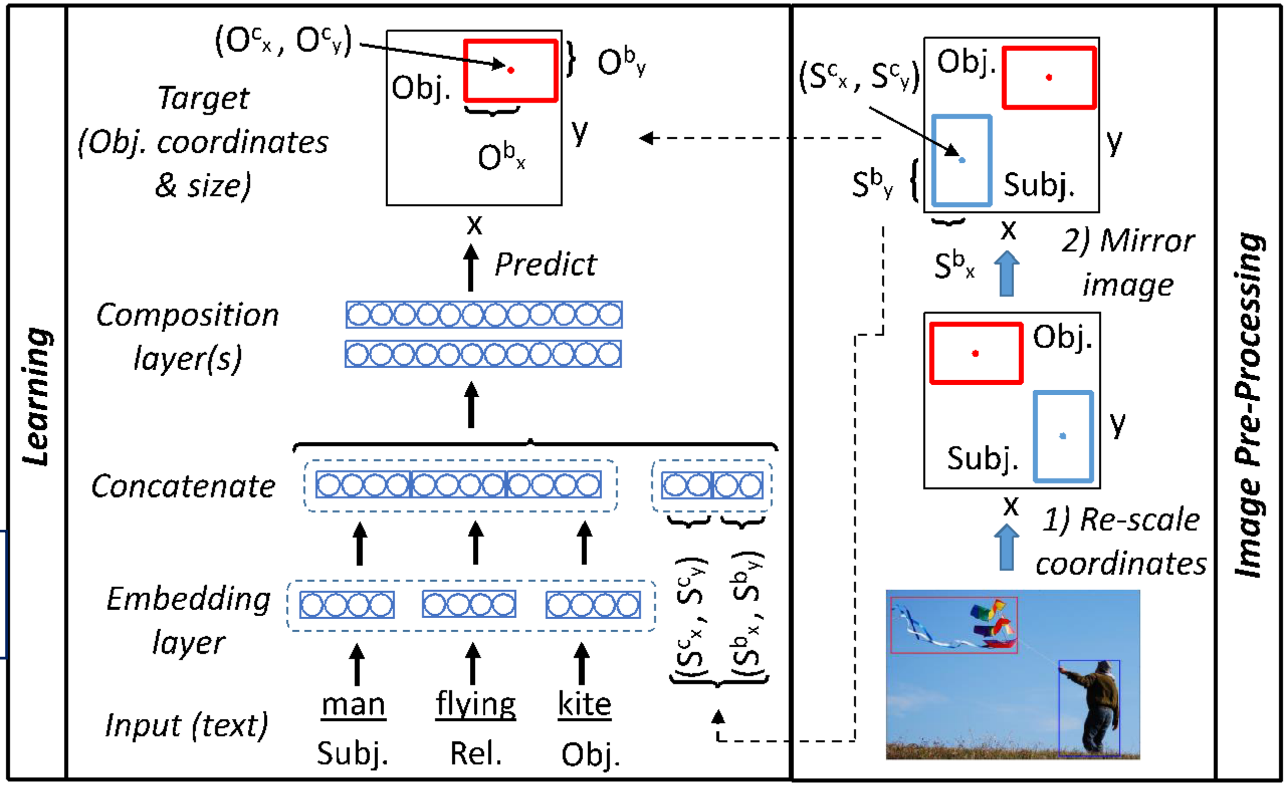
Guillem Collell & Marie-Francine Moens (2018). Learning representations specialized in spatial knowledge: Leveraging language and vision. *Transactions of the Association for Computational Linguistics* (TACL), 6, 133-144.

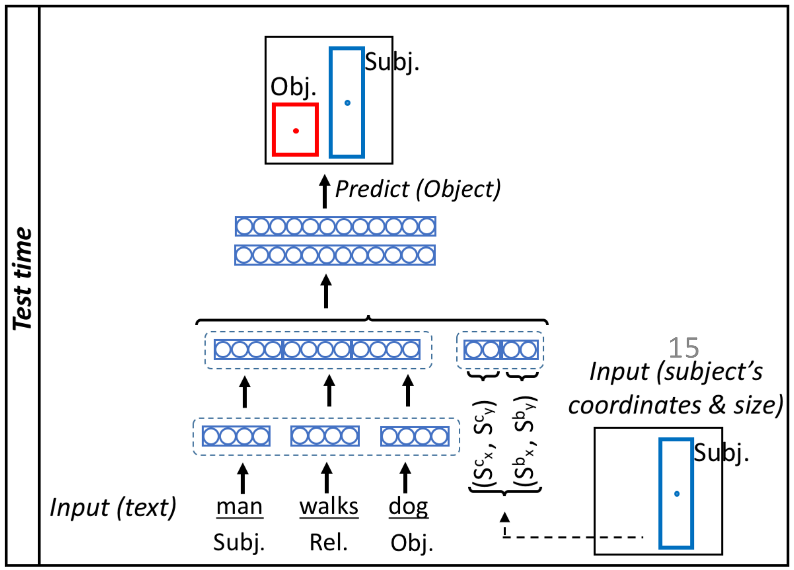
Simple feedforward neural network

Loss: mean squared error

Word embeddings to generalize over unseen words

Triplet of words, coordinates of subject





Guillem Collell, Luc Van Gool & Marie-Francine Moens (2018). Acquiring common sense spatial knowledge through implicit spatial templates. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence* (AAAI 2018) (pp. 6765-6772). AAAI.

Visualizing the location of an object

Quantitative evaluation: 10-fold cross-validation and results averaged over the 10 folds

		MSE	R^2	acc_y	$F1_y$	r_x	r_y
Implicit	*EMB*	**0.008**	0.705	0.756	0.755	0.894	0.834
	RND	**0.008**	0.691	0.750	0.750	0.891	0.826
	1H	**0.008**	**0.717**	**0.762**	**0.762**	**0.896**	**0.842**
	ctrl	0.054	-1.000	0.522	0.521	0.000	-0.001
Explicit	*EMB*	0.013	0.586	0.768	0.770	0.811	0.823
	RND	0.013	0.580	0.767	0.769	0.808	0.815
	1H	**0.012**	**0.604**	**0.778**	**0.780**	**0.815**	**0.828**
	ctrl	0.060	-1.000	0.633	0.630	0.000	0.000

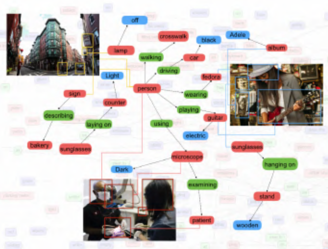
Table 1: Results on **implicit** and **explicit** relations.

EMB: Glove embeddings as input

RND: Random embeddings as input

1H: 1-hot encodings as input

Ctrl: control method that outputs random normal predictions



Visual Genome data set: 108K images with 1,5M human-annotated (Subject, Relationship, Object) instances with bounding boxes for Subject and Object

Visualizing the location of an object

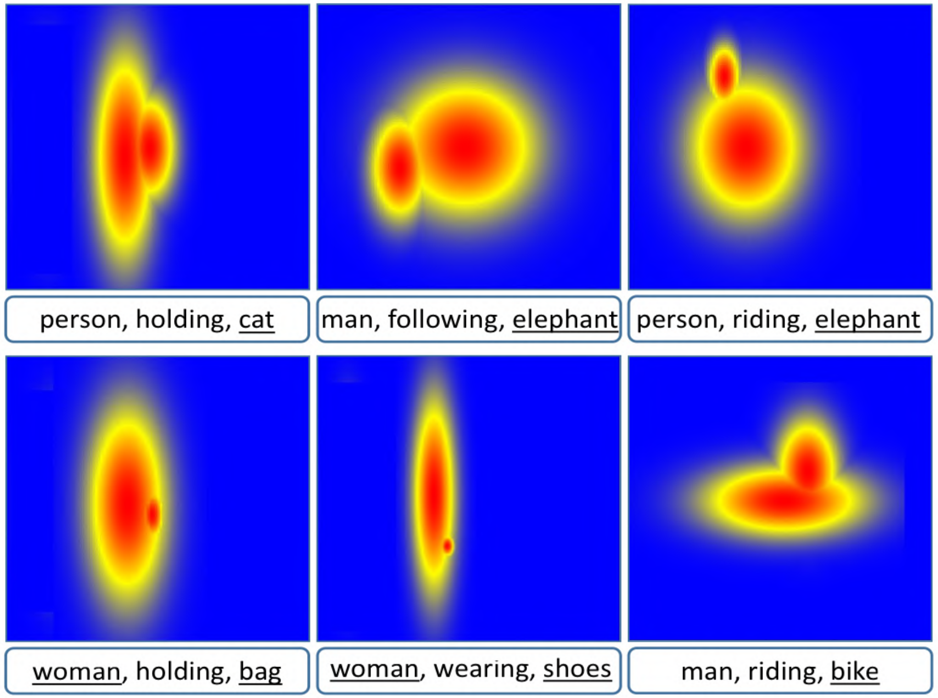
Qualitative evaluation

Figure 2: Predictions by the model that leverages word embeddings (*EMB*). **Top**: Predictions in unseen words (underlined). **Bottom:** Predictions in unseen *triplets*.

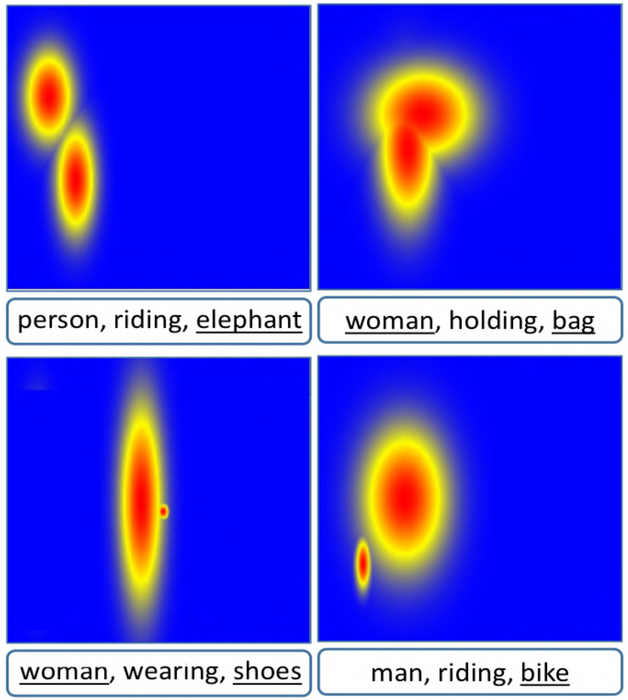
Guillem Collell & Marie-Francine Moens (2017). Learning Visually Grounded Common Sense Spatial Knowledge for Implicit Spatial Language. In *Proceedings of the 13th International Symposium on Commonsense Reasoning*, University College London. CEUR.

Visualizing the location of an object

Qualitative evaluation

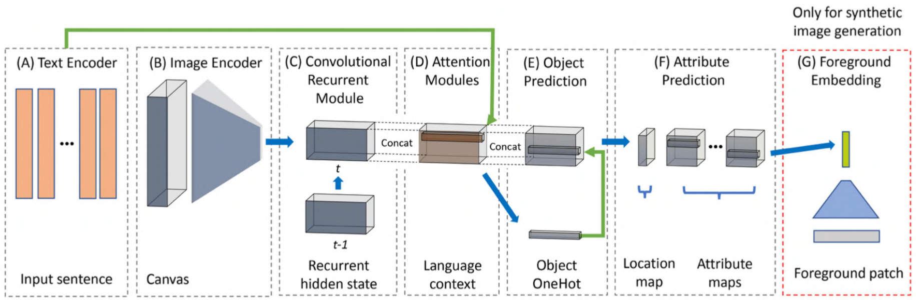


Model: Initialized with distributional word embeddings



Model: Initialized with random word embeddings

Text to scene translation



- **Attention based object decoder**:
  - Outputs the likelihood scores of all possible objects in the object vocabulary $\mathcal{V}$
  - Uses the recurrent scene state h_t^S, text features $\{[h_i^E, x_i]\}$, and the previously predicted object o_{t-1}
- **Attention based attribute decoder**

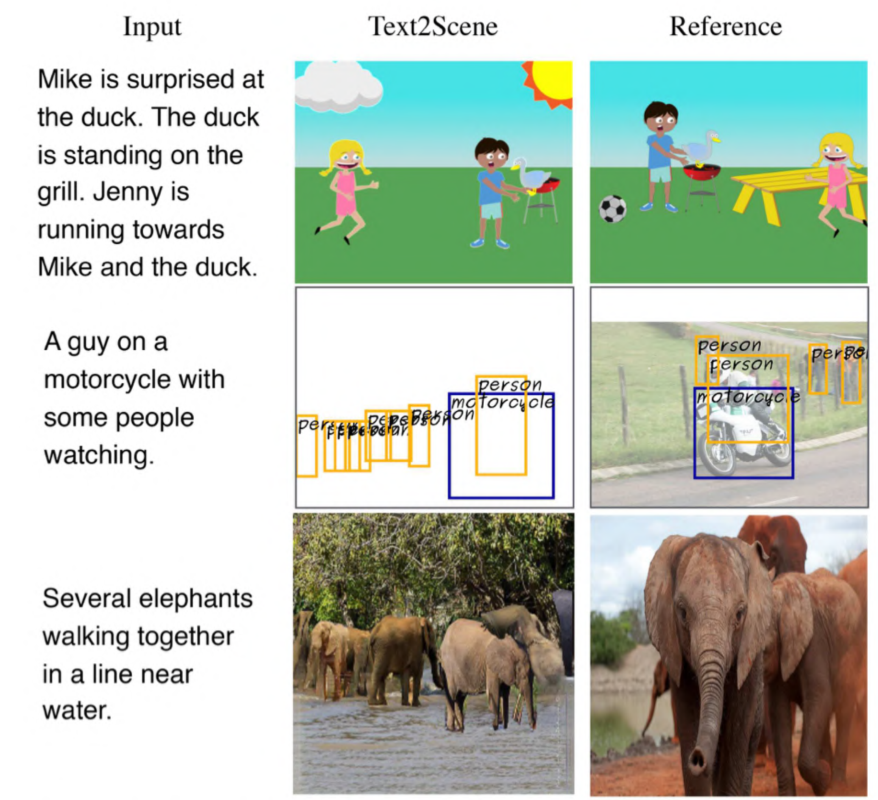


Figure 1. Sample inputs (left) and outputs of our Text2Scene model (middle), along with *ground truth* reference scenes (right) for the generation of abstract scenes (top), object layouts (middle), and synthetic image composites (bottom).

Fuwen Tan, Song Feng & Vincente Ordonez (2018). Text2Scene: Generating compositional scenes from textual descriptions. In *Proceedings of the Conference on Computer Vision and Pattern Recognition* (CVPR).

Text to scene translation

More details

$$u_t^o = \text{AvgPooling}\left(\Psi^o(h_t^S)\right)$$

Ψ^o = CNN with spatial attention on h_t^S to collect the spatial context about the objects already added; attended spatial features are then fused by average pooling forming vector u_t^o

$$c_t^o = \Phi^o\left([u_t^o; o_{t-1}], \left\{\left|h_i^E, x_i\right|\right\}\right)$$

Φ^o = text-based attention module, which uses $[u_t^o; o_{t-1}]$ to attend to the language content $\left\{\left|h_i^E, x_i\right|\right\}$ resulting in context vector c_t^o

$$P(o_t) \propto \Theta^o([u_t^o; o_{t-1}; c_t^o])$$

Θ^o= a two-layer perceptron that predicts the likelihood of the next object from the concatenation of u_t^o, o_{t-1}, and c_t^o using a softmax function

Trained with negative log-likelihood losses corresponding to the object, location, and discrete attribute softmax classifiers

Fuwen Tan, Song Feng & Vincente Ordonez (2018). Text2Scene: Generating compositional scenes from textual descriptions. In *Proceedings of the Conference on Computer Vision and Pattern Recognition* (CVPR).

Text to scene translation

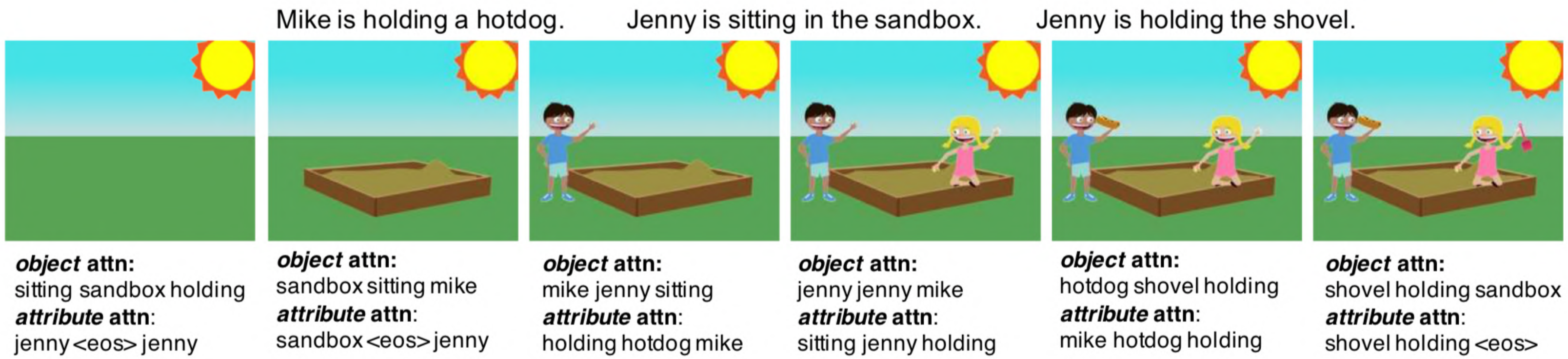


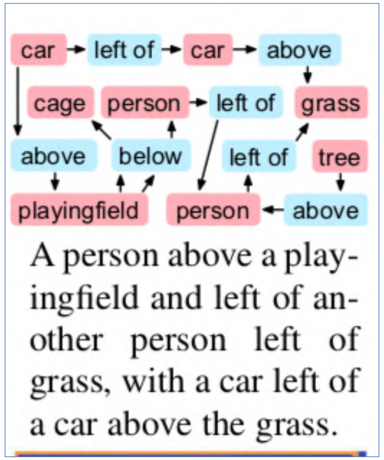
Figure 3. Step-by-step generation of an abstract scene, showing the top-3 attended words for the object prediction and attribute prediction at each time step. Notice how except for predicting the *sun* at the first time step, the top-1 attended words in the object decoder are almost one-to-one mappings with the predicted objects. The attended words by the attribute decoder also correspond semantically to useful information for predicting either pose or location, e.g. to predict the location of the *hotdog* at the fifth time step, the model attends to *mike* and *holding*.

Fuwen Tan, Song Feng & Vincente Ordonez (2018). Text2Scene: Generating compositional scenes from textual descriptions. In *Proceedings of the Conference on Computer Vision and Pattern Recognition* (CVPR).

Text to scene translation: integration of a scene graph

- Text is first translated into a scene graph (= symbolic representation expressing the objects and their semantic/spatial relationships)

- The spatial layout is generated from the scene graph

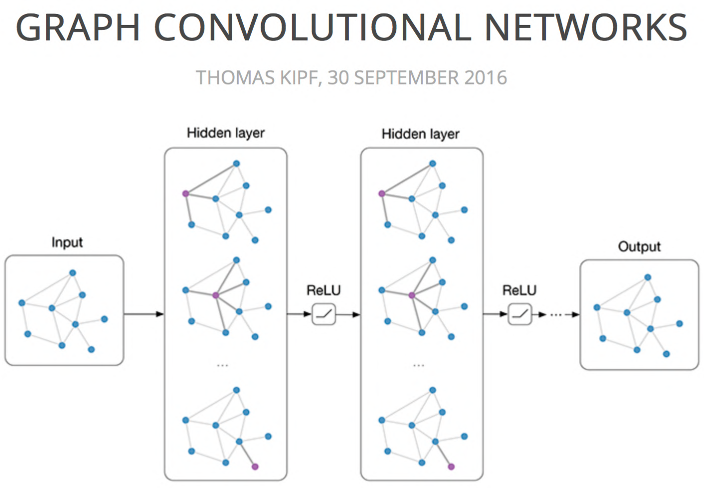


- Use of a graph convolution network composed of several graph convolution layers to represent objects and their relationships

- Followed by steps of layout prediction and pixel prediction

Graph convolution network

- Graph convolution network:
  - Input: graph with vectors of dimension D_{in} at each node and edge, it computes new vectors of dimension D_{out} for each node and edge => graph convolution propagates information along edges of the graph
  - Can be seen as a message passing algorithm where, e.g., the representation of a node is updated based on "messages" sent by neighboring nodes



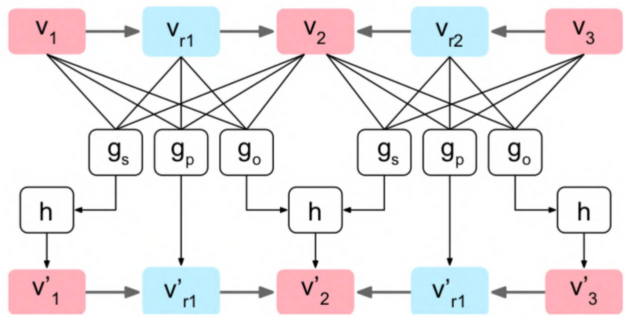
Multi-layer Graph Convolutional Network (GCN) with first-order filters.

Graph convolution network

Given input vectors $v_i, v_r \in \mathbb{R}^{D_{in}}$ for all objects $o_i \in O$ and edges $(o_i, r, o_j) \in E,$ we compute output vectors for $v_i', v_j' \in \mathbb{R}^{D_{out}}$ for all nodes and edges using three functions g_s, g_p and g_o, which take as input the triple of vectors (v_i, v_r, v_j) for an edge and output new vectors for the subject o_i, predicate r, and object o_j, respectively

Output:

- $v_r' = g_p(v_i, v_r, v_j)$
- An object may participate in many relationships:
  - v_i' depends on all vectors v_j for objects to which o_i is connected via graph edges as well as the vectors v_r for those edges
  - $V_i^s = \{g_s(v_i, v_r, v_j) : (o_i, r, o_j) \in E\}$
  - $V_i^o = \{g_o(v_j, v_r, v_i) : (o_j, r, o_i) \in E\}$
  - v_i' for object o_i is then computed as $v_i' = h(V_i^s \cup V_i^o)$ where h is a symmetric function which pools an input set of vectors to a single output vector



Justin Johnson, Agrim Gupta & Li Fei-Fei (2018). Image generation from scene graphs. In *Proceedings of the Conference on Computer Vision and Pattern Recognition* (CVPR).

Text to scene translation: integration of a scene graph

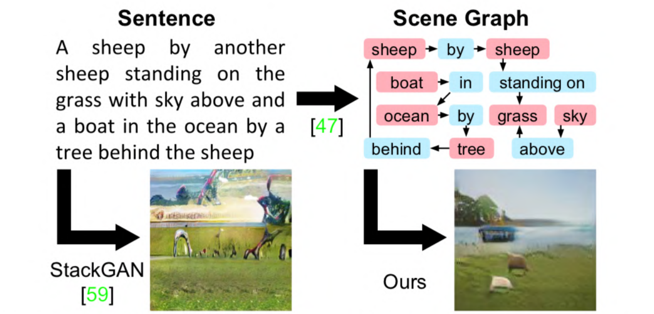


Figure 1. State-of-the-art methods for generating images from sentences, such as StackGAN [59], struggle to faithfully depict complex sentences with many objects. We overcome this limitation by generating images from *scene graphs*, allowing our method to reason explicitly about objects and their relationships.

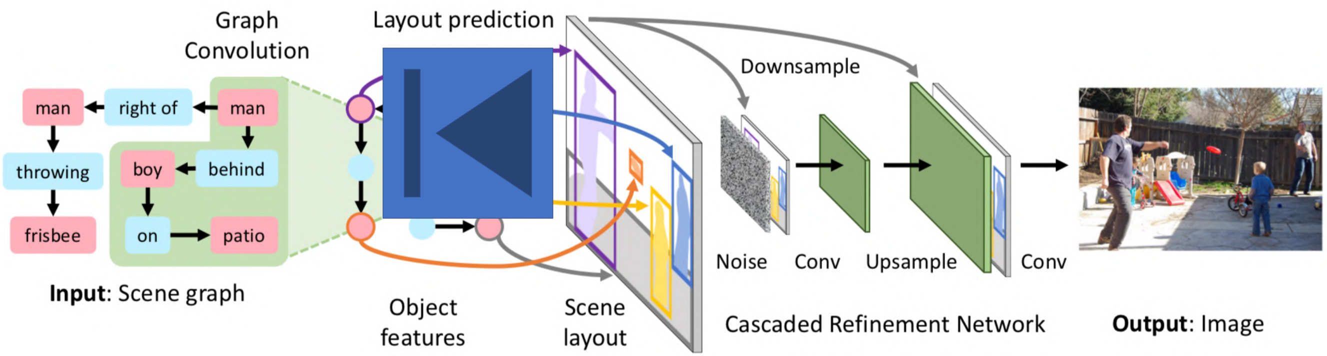


Figure 2. Overview of our image generation network f for generating images from scene graphs. The input to the model is a *scene graph* specifying objects and relationships; it is processed with a *graph convolution network* (Figure 3) which passes information along edges to compute embedding vectors for all objects. These vectors are used to predict bounding boxes and segmentation masks for objects, which are combined to form a *scene layout* (Figure 4). The layout is converted to an image using a *cascaded refinement network* (CRN) [6]. The model is trained adversarially against a pair of *discriminator networks*. During training the model observes ground-truth object bounding boxes and (optionally) segmentation masks, but these are predicted by the model at test-time.

Justin Johnson, Agrim Gupta & Li Fei-Fei (2018). Image generation from scene graphs. In *Proceedings of the Conference on Computer Vision and Pattern Recognition* (CVPR).

Text to scene translation: integration of a scene graph

More details

- Scene graphs were manually created, but could be derived from dependency parse
- A generative adversarial network was trained end-to-end including several loss functions
- Interesting to mention is the box loss for layout prediction:
  - *Box loss:* $\mathcal{L}_{box} = \sum_{i=1}^{n} \left\| b_i - \hat{b}_i \right\|_1$ which penalizes the L_1 difference between ground-truth b_i and predicted box $\hat{b}_i$, where n = number of objects in the graph
  - Optimized over all N training data

- Problem of semantic standards for object and relationship names in the scene graph

Justin Johnson, Agrim Gupta & Li Fei-Fei (2018). Image generation from scene graphs. In *Proceedings of the Conference on Computer Vision and Pattern Recognition* (CVPR).

Text to scene translation

The LSTM encoder provides a representation (embedding) of each object mentioned in the input text

From this representation a bounding box of the object is predicted in the 2D space:
The output of the box generator is a set of bounding boxes **B** $= \{B_1, \ldots, B_n\}$ where each bounding box B_t defines the location, size and category label of the t-th object

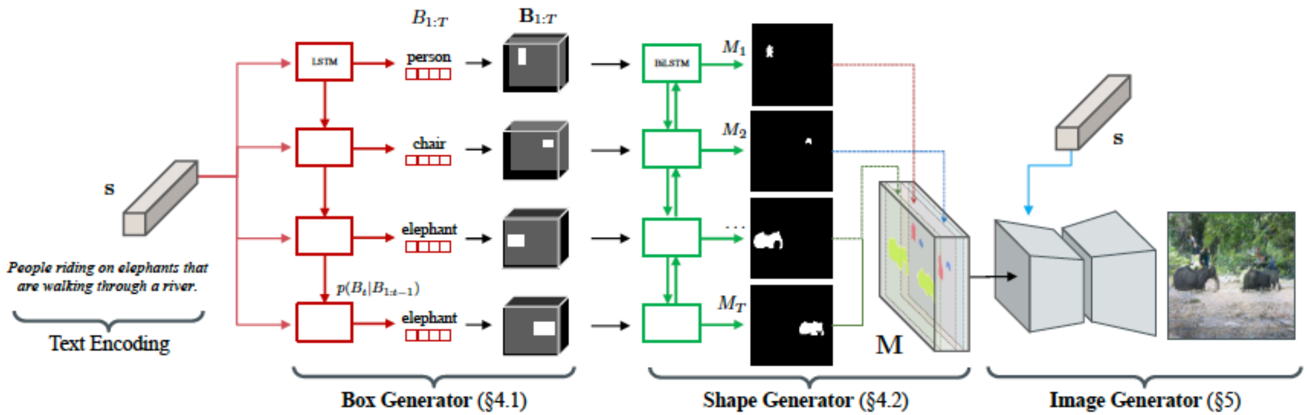


Figure 2. Overall pipeline of the proposed algorithm. Given a text embedding, our algorithm first generates a coarse layout of the image by placing a set of object bounding boxes using the box generator (Section 4.1), and further refines the object shape inside each box using the shape generator (Section 4.2). Combining outputs from the box and the shape generator leads to a semantic label map defining semantic structure of the scene. Conditioned on the inferred semantic layout and the text, a pixel-wise image is finally generated by the image generator (Section 5).

Seunghoon Hong, Dingdong Yang, Jongwook Choi (2018). Inferring semantic layout for hierarchical text-to-image synthesis. In *Proceedings of the Conference on Computer Vision and Pattern Recognition* (CVPR).

Text to scene translation

More details

- We denote the labeled bounding box of the t-th object as $B_t = (\mathbf{b}_t, \boldsymbol{l}_t)$, where $\mathbf{b}_t = [b_{t,x}, b_{t,y}, b_{t,w}, b_{t,h}]$ $\in \mathbb{R}^4$ = the location and size of the bounding box, and $\boldsymbol{l}_t \in \{0,1\}^{L+1}$ is a one-hot class label over L categories; $(L+1)$-th class as a special indicator for the end-of-sequence
- Bounding box generator = auto-regressive (i.e., it uses prediction from a previous state to generate next step) decoder modeled by decomposing the joint conditional box probability as $P(\mathbf{B}_{1:n}|\mathbf{s})$ $= \prod_{t=1}^{n} P(B_t|B_{1:t-1}, \mathbf{s})$ where $\mathbf{s}$ is the input text
- We first sample the class label $\boldsymbol{l}_t$ for the t-th object and then generate the box coordinates $\mathbf{b}_t$ conditioned on $\boldsymbol{l}_t$, i.e., $P(B_t| \cdot) = P(\boldsymbol{l}_t, \mathbf{b}_t| \cdot)$= $P\ (\boldsymbol{l}_t \mid \cdot)\ P\ (\mathbf{b}_t|\boldsymbol{l}_t, \cdot)$
- Training by minimizing the negative log-likelihood of ground-truth bounding boxes and their labels:

$$\mathcal{L}_{box} = -\lambda_l \frac{1}{n}\sum_{t=1}^{n} \boldsymbol{l}_t^* \log P(\boldsymbol{l}_t) - \lambda_b \frac{1}{n}\sum_{t=1}^{n} \log P(\mathbf{b}_t^*)$$

optimized over all N training data

Seunghoon Hong, Dingdong Yang, Jongwook Choi (2018). Inferring semantic layout for hierarchical text-to-image synthesis. In *Proceedings of the Conference on Computer Vision and Pattern Recognition* (CVPR).

Text to scene translation

- Qualitative evaluation of the full image generation process

Seunghoon Hong, Dingdong Yang, Jongwook Choi (2018). Inferring semantic layout for hierarchical text-to-image synthesis. In *Proceedings of the Conference on Computer Vision and Pattern Recognition* (CVPR).

Transformer for spatial language modeling

- Famous for language modeling: e.g., BERT: Bidirectional Encoder Representations from Transformers and variants
- Increasingly popular for jointly modeling language and visual data: e.g., LXMERT, VilBERT, VLBERT, etc. for better understanding of language and visual data (e.g., in visual question answering, visual dialog)
- Is this architecture also suited to model spatial language?

Jacob Devlin, Ming-Wei Chang, Kenton Lee & Kristina Toutanova. (2018). BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies* (pp. 4171-4186). ACL.

Spatial-Reasoning BERT

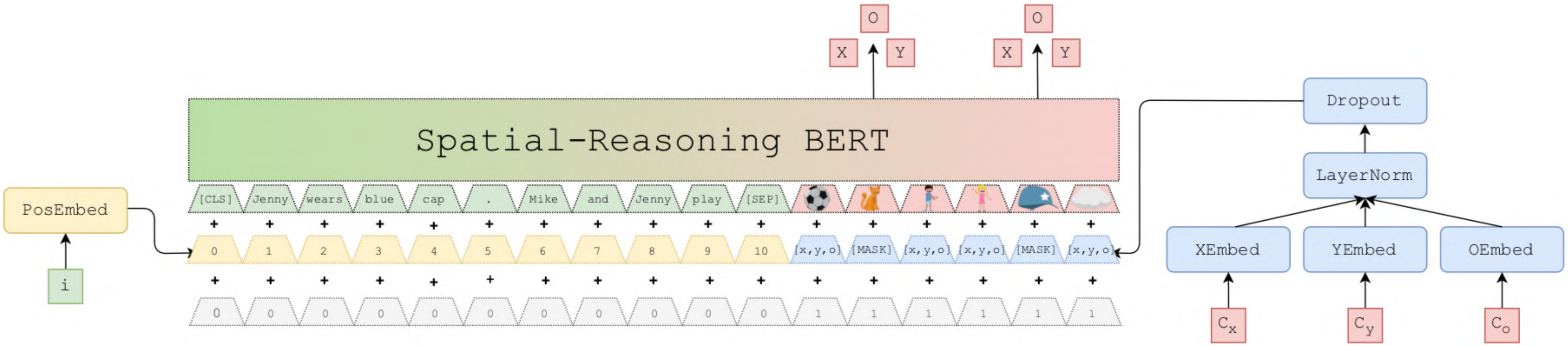


Figure 2: The SR-BERT backbone architecture with the text position embedding module as per BERT – Left (Yellow), clip-art spatial embedding module, which is novel in SR-BERT – Right (Blue). The blue [MASK] elements are the masked spatial positions, which the model learns to predict during training. During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

Model is trained by minimizing the sum of the individual per-axis cross-entropy losses $\mathcal{L}_x$ and $\mathcal{L}_y$ together with the orientation loss $\mathcal{L}_{or}$

Gorjan Radevski, Guillem Collell, Marie-Francine Moens & Tinne Tuytelaars (2020). Decoding language spatial relations to 2D spatial arrangements. *EMNLP Findings*.

Spatial-Reasoning BERT

Quantitative and qualitative evaluation

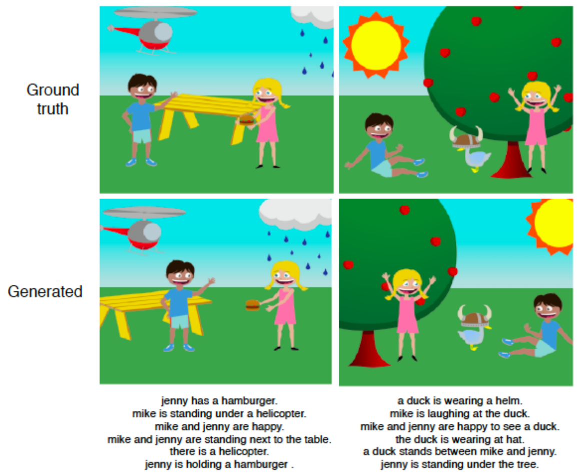


Figure 1: Ground truth and generated spatial arrangements expressed as 2D scenes of clip-arts. For each scene, there is a set of spatial relationships expressed in both implicit and explicit language.

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).

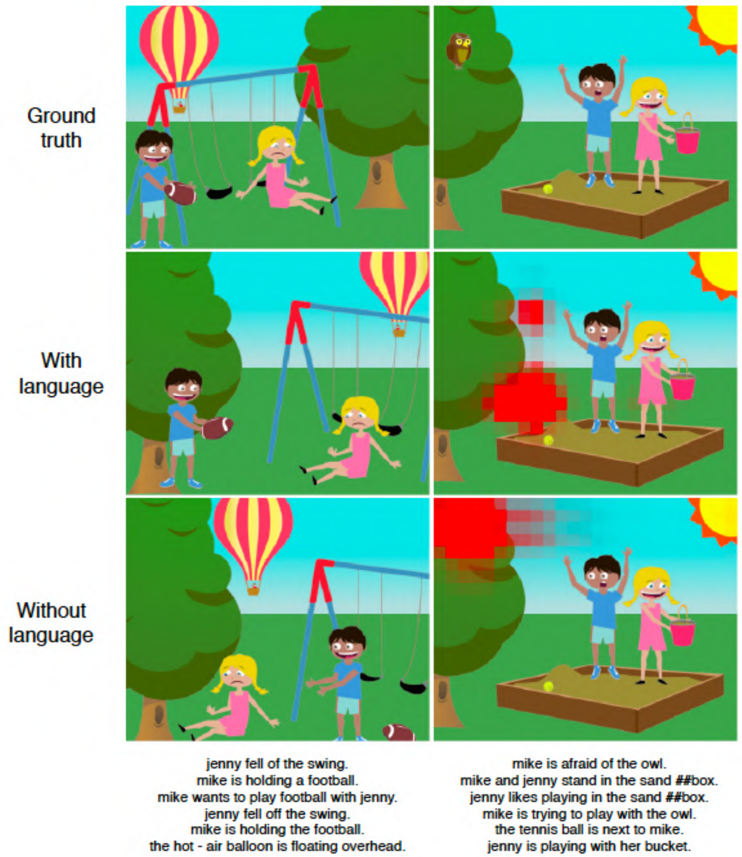
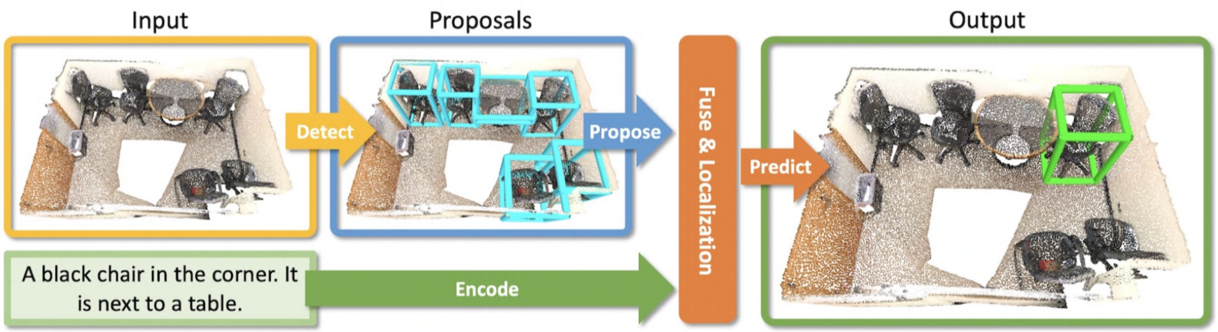


Figure 4: Ground truth (top), generated scenes (middle-left, bottom-left) and heat-maps (middle-right, bottom-right) with and without conditioning on the language.

Gorjan Radevski, Guillem Collell, Marie-Francine Moens & Tinne Tuytelaars (2020). Decoding language spatial relations to 2D spatial arrangements. *EMNLP Findings*.

Scene layout generation in 3D-space

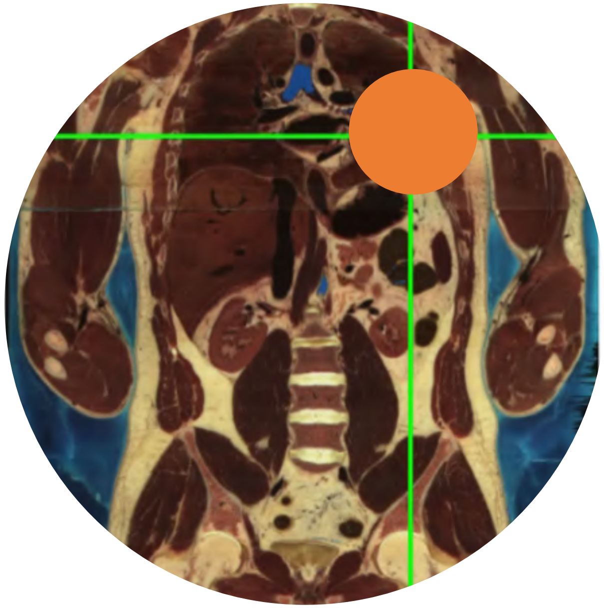
- Task of object localization using natural language directly in 3D space
  - Input is given text description
  - Output: Predict position of referred object in the 3D scene



Dave Zhenyu Chen, Angel X. Chang & Matthias Niessner (2020). ScanRefer: 3D object localization in RGB-D scans using natural language. In *Proceedings of the European Conference on Computer Vision* (ECCV).

Map text to 3D

- Representations of organs mentioned in medical text is projected to location in 3D atlas of the human body
- Embeds medical text into a universal, small dimensional space corresponding to the human body that is easy to navigate and interpret
- The volume of each organ is characterized by a set of voxels in the atlas, which capture its position, size and shape
- The voxels of one organ can, in turn, be represented by a point cloud in 3D space, where each point represents the coordinate indices of one voxel

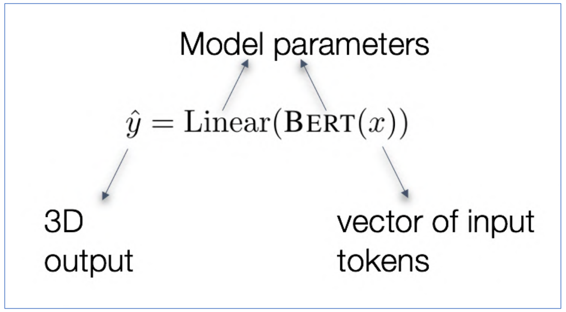


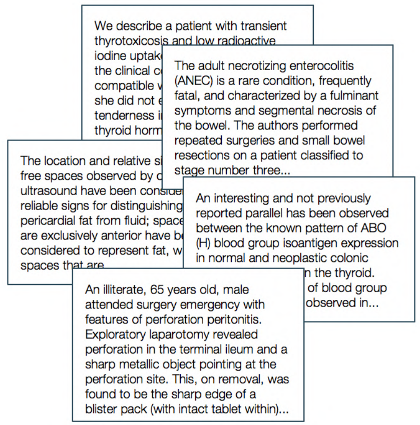
Task: **Grounding medical text** in the human body

Dusan Grujicic, Gorjan Radevski, Tinne Tuytelaars & Matthew Blaschko (2020). Learning to ground medical text in a 3D human atlas. In *Proceedings of the Conference on Computational Natural Language Learning* (CoNLL). ACL.

Map text to 3D

- BERT backbone
- Model input — Medical text tokenized with WordPiece
- Model output — [CLS] token representation projected into 3D





- Loss function: Enables **reasoning about** the semantic relatedness of medical text

Dusan Grujicic, Gorjan Radevski, Tinne Tuytelaars & Matthew Blaschko (2020). Learning to ground medical text in a 3D human atlas. In *Proceedings of the Conference on Computational Natural Language Learning* (CoNLL). ACL.

Map text to 3D

Soft Organ Distance loss

- Not only grounding the medical article to the right organ but also to the appropriate location within the organ based on the other organs mentioned as context without any explicit annotations at that level of granularity
- Could be refined by considering spatial language

Total number of organs

$$\mathcal{L}_t = \sum_{i=1}^{M} \mathcal{L}_o^i \frac{\exp(-\mathcal{L}_o^i/\gamma_o)}{\sum_{j=1}^{M} \exp(-\mathcal{L}_o^j/\gamma_o)}$$

Total loss minimized

Organ loss for *i*-th organ calculated as the sum of contributions of its points

Temperature term

Softmin across the distances as weights for the contributions of individual points

Euclidean distances between the prediction & each sampled organ point

Loss contribution of i-th organ point

$$\mathcal{L}_p = \|\hat{y} - y\|_2 \frac{\exp(-\|\hat{y} - y\|_2/\gamma_p)}{\sum_{i=1}^{N} \exp(-\|\hat{y} - y_i\|_2/\gamma_p)} \qquad \mathcal{L}_o = \sum_{i=1}^{N} \mathcal{L}_p^i$$

Loss contribution of an organ point

Model prediction

Organ point

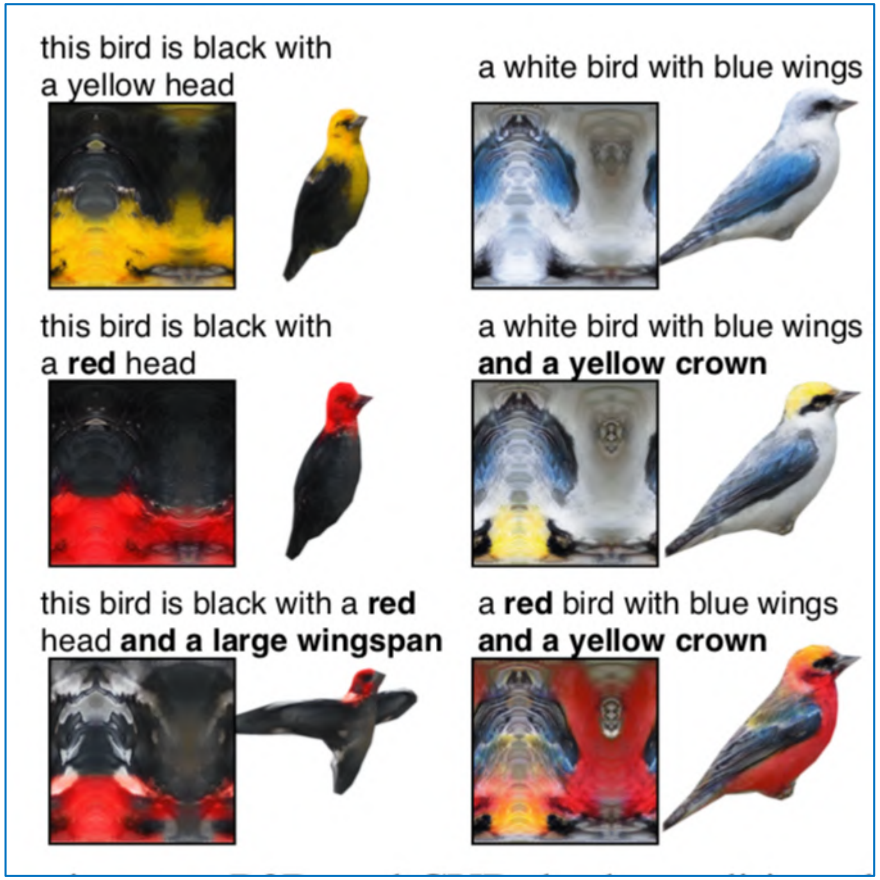
Temperature term

Loss contribution of organ

Dusan Grujicic, Gorjan Radevski, Tinne Tuytelaars & Matthew Blaschko (2020). Learning to ground medical text in a 3D human atlas. In *Proceedings of the Conference on Computational Natural Language Learning* (CoNLL). ACL.

Control 3D with language

- The goal is to gain more control in GAN based image generation
- Natural disentanglement of shape and color in the image generation process
- The methodology maps the 3D shapes in 2D space so that they are pose-independent (i.e., the beak, tail, wings are always in the same location)
- This makes it easier for the attention mechanisms to map the language information to the visual space and control the image generation



Dario Pavllo, Graham Spinks, Thomas Hofmann, Marie-Francine Moens & Aurelien Lucchi (2020). Convolutional Generation of Textured 3D Meshes. In *Advances in Neural Information Processing Systems Volume* 33.

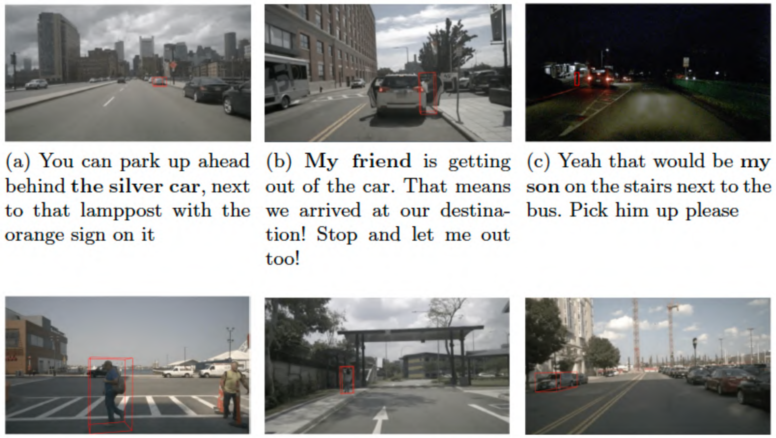
Application: Giving a command to your self-driving car

C4AV @ ECCV 2020

COMMANDS FOR AUTONOMOUS VEHICLES WORKSHOP
23 AUGUST 2020 - GLASGOW

Challenge

- The task of visual grounding requires locating the most relevant region or object in an image, given a natural language query.



(a) You can park up ahead behind **the silver car**, next to that lamppost with the orange sign on it

(b) **My friend** is getting out of the car. That means we arrived at our destination! Stop and let me out too!

(c) Yeah that would be **my son** on the stairs next to the bus. Pick him up please

(d) After **that man in the blue top** has passed, turn left

(e) There's **my mum**, on the right! The one walking closest to us. Park near **her**, she might want a lift

(f) Turn around and park in front of **that vehicle in the shade**

Fig. 2: Some examples from the Talk2Car dataset [5]. Each command describes an action that the car has to execute relevant to a referred object found in the scene (here indicated by the red 3D-bounding box). The referred object is indicated in bold in each command. Best seen in color.

Thierry Deruyttere, Simon Vandenhende, Dusan Grujicic, Yu Liu, Luc Van Gool, Matthew Blaschko, Tinne Tuytelaars & Marie-Francine Moens (2020). Commands 4 Autonomous Vehicles (C4AV) Workshop Summary. In *Proceedings of the 16th European Conference on Computer Vision.*

Application: Giving a command to your self-driving car

C4AV @ ECCV 2020

COMMANDS FOR AUTONOMOUS VEHICLES WORKSHOP
23 AUGUST 2020 - GLASGOW

- Best results in terms of IoU by using Stacked VLBert model

Model	AP_{50}	Parameters (M)	Inference Speed (ms)
Stacked VLBert	**0.710**	683.80	240.79
MMT	0.691	194.97	125.50
Third Place	0.686	366.50	74.44
ASSMR	0.660	**48.91**	**47.23**
One-Stage Grounding	0.603	75.40	123.12
MSRR [4]	0.601	62.25	270.50
MAC [12]	0.505	41.59	51.23
Inner-Product Model [24]	0.441	15.80	10.24

Table 2: The results on the Talk2Car test set. The models under the line in the middle of the table are baseline models. Inference speed was measured on a Nvidia RTX Titan.

Vision	Language	Word Attention
ResNet-50	BERT	Yes
ResNet-152	Transformer	Yes
EfficientNet	Sentence-Transformer [20]	Yes
ResNet-18	GRU	Yes
DarkNet-53	RNN	Yes
ResNet-101	LSTM	Yes
ResNet-101	LSTM	Yes
ResNet-18	LSTM	No

Thierry Deruyttere, Simon Vandenhende, Dusan Grujicic, Yu Liu, Luc Van Gool, Matthew Blaschko, Tinne Tuytelaars & Marie-Francine Moens (2020). Commands 4 Autonomous Vehicles (C4AV) Workshop Summary. In *Proceedings of the 16th European Conference on Computer Vision.*

Reasoning in physical space

- The above approaches allow reasoning in the physical space (2D or 3D): is useful in processing human-machine communications: e.g.,
  - Communications with robots and autonomous vehicles: inference of additional spatial information
  - Still large potential for learning from video data coupled with language

- When to reason in the language space (use of spatial ontology) and when in the physical space is an interesting research question

- Both methods are transparent for humans and contribute to the explainability of the models

Reasoning in representation space that mimics the human brain

- Representations that generate the mappings of language to 2D or 3D spaces contain the spatial information in a dense, distributed form

- Eventually quantitative spatial reasoning with these ???
- Inspired by the human brain?
- Possibly computations in non-Euclidean geometric spaces ???