Supplementary Materials for Unified Visual-Semantic Embeddings

This supplementary material is organized as follows. First, in Appendix A, we provide more details for the implementation of our model and the training method. Second, in Appendix B, we provide the experiment setups, metrics, baseline implementations, qualitative examples and analysis for each experiment we discussed in the main text. We end this section with the visualization of the learned unified VSE space of different semantic levels.

A. Implementation Details

A.1. Generating Negative samples

To generate negative samples in sentence level, we follow the sampling paradigm introduced by [1]: We sample negative examples from all other captions/images in the dataset in a training batch. Note that as [1] shown, the batch size will largely affect the models' performance. For a fair comparison, we set the batch size as 128 which is the same as [1, 4]. In the rest of this section, we discuss in detail how we sample negative semantic components.

As for nouns, we sample 16 negative nouns from a fixed set of nouns: This noun set is extracted from nouns with frequency more than 100 (in total 1,205 nouns extracted in MS-COCO dataset).

As for attribute-noun pairs, we randomly sample 8 other attributes in a fixed attribute set and replace the original attribute in the pair, as negative examples. The attribute set is composed by the frequently appeared attributes in the MS-COCO dataset. In detail, we extract in total 37 attributes, *i.e.*, `white`, `black`, `red`, `green`, `brown`, `yellow`, `orange`, `pink`, `gray/grey`, `purple`, `young`, `wooden`, `old`, `snowy`, `grassy`, `cloudy`, `colorful`, `sunny`, `beautiful`, `bright`, `sandy`, `fresh`, `morden`, `cute`, `dry`, `dirty`, `clean`, `polar`, `crowded`, `silver`, `plastic`, `concrete`, `rocky`, `wooded`, `messy`, `square`. We also randomly replace nouns in the pairs to generate another set of negative attribute-noun pairs. For each attribute-noun pair, we randomly draw 16 negative examples.

We separately compute the ranking loss corresponding to two types of negatives, denoted as $\ell_{attr_{negnoun}}$ and $\ell_{attr_{negattr}}$. Both of them are computed by a uni-directional ranking loss with negative examples drawn in text-domain. OHEM strategy is not applied on them. The final loss is the sum of them, *i.e.*, $\ell_{attr} = \ell_{attr_{negnoun}} + \ell_{attr_{negattr}}$.

Here we add a small note for the reproducibility. In cases with multiple modifiers on the nouns (*e.g.*, `old black dog`), for simplicity, in our implementation, we always extract the first modifier of each noun phrases as its attribute (`old dog` in this case).

As for relational triples, we randomly sample 4 relational words and 2 negatives subjects (nouns) and 2 negative objects (nouns) to replace the corresponding parts in the triple, as negative examples. In total, we have 8 negative triples for each relational triple The choice of this small number of negative examples is attributed to the trade-off between the computational efficiency and stability of training. Empirically, we find that increasing the number of negative triples does not bring much improvement to the performance.

We also sample negative relational triples from other captions within the training batch. In detail, we sample 1 negative rational triple for each other caption within the batch. This results in at most 128-1 = 127 negative examples for each relational triple ("at most" means some captions may not contain relational phrases). Similar as attribute-noun pairs, we individually compute the ranking loss on each type of negatives and sum them together as the ℓ_{rel}. The losses are computed by uni-directional ranking loss without OHEM.

As for negative bag-of-components, we sample negative ones in a similar manner as we do for sentences: We draw them from the bag-of-components in other captions within the training batch. We also draw other images from batch as negative images. The loss ℓ_{comp} is computed by bi-directional ranking loss with OHEM strategy.

A.2. Model settings and details

Weight of $\boldsymbol{\eta_c}$, $\boldsymbol{\eta_o}$, $\boldsymbol{\eta_a}$, $\boldsymbol{\eta_r}$. The choice of the 4 hyperparameters in Eq.(4) (*i.e*., η_c, η_o, η_a, η_r) in the main text actually has no significant influence on the model's performance, as they all contribute to the better alignment between two modalities. To show this, we fix three of the ηs and test 5 different values for the rest one (*e.g*., set $\eta_c \in \{0.1, 1, 2, 4, 8\}$). The bidirectional retrieval scores `rsum` of all 20 models are within the range of 468.2 ± 2.

Dependency on the semantic parser. Recall that the semantic components are all extracted by the semantic parser and we evaluate the influence of the recall of the semantic parser on the model's performance by randomly dropping 30% relations and 30% attributes from the parser's output during training/test. Shown in Table 1, the recall of the parser on training captions has a small contribution to the performance. However, low recalls on test captions noticeably degenerate the performance on discriminating adversarial captions, because UniVSE relies on the parsed components to find unmatched components between images and texts. Thus, in Section 4.4, we show that UniVSE can facilitate the semantic parsing with visual cues.

Spatial aggregation method. For the spatial aggregation $\Psi(\cdot)$ of the 7×7 image feature maps, instead of using the max pooling which may drop most information in the feature map or the average pooling which tends to include noises, we adopt a specific pooling method called max-k pooling. Max-k pooling select k largest values in the feature map and return the average value of these k largest responses. Formally, for the feature map $\mathbf{V} \in \mathbb{R}^{7\times7\times d}$, denoting the k-th largest value in i-th channel of $\mathbf{V}$ as $\mathbf{V}_k[i]$, the max-k pooled global image embedding $\mathbf{v}$ can be formalized as $\mathbf{v}[i] = \mathrm{mean}(\{\mathbf{V}[x, y, i] | \mathbf{V}[x, y, i] \leq \mathbf{V}_k[i], x, y \in 7 \times 7\})$. Obviously, max pooling is the specific form of max-k pooling when $k = 1$ and average pooling can be regarded as max-49 pooling (for a 7×7 spatial resolution). In the experiments, we empirically set k to 10, as a trade-off between removing useful information and retaining unimportant information. Table 2 shows the performance of different spatial pooling methods. We can observe that the proposed max-k pooling achieves best performance among three pooling methods (*i.e*., average pooling, max pooling and max-k pooling). Notice that, max-k pooling will bring better performances compared with max/average pooling, however, the max-k pooling has to be trained under UniVSE structure (*i.e*., trained with $\ell_{comp}, \ell_{obj}, \ell_{attr}, \ell_{rel}$), otherwise, the local correspondences will not be learned well. The results in Table 2 shows a significant performance drop on defending the adversarial captions, if UniVSE (with max-10 pooling) is trained without loss ℓ_{comp}.

Semantic aggregation method For the aggregation function $\Phi(\cdot)$ for semantic components to generate $\mathbf{u}_{comp}$, we have added some experiments. Specifically, we evaluate the performance of both hand-coded functions: average pooling, sum, and max pooling (by taking a channel-wise max of all components), as well as learnable functions: GRU and self-attentive pooling [2]. For the GRU alternative, we treat the set of components as a sequence (ordered randomly), encode it with a GRU module, and use the last hidden state as the $\mathbf{u}_{comp}$. The results are summarized in Table 3. GRU performs slightly better than other methods on the standard retrieval task. However, it requires extra computation cost. Max pooling outperforms others in discriminating adversarial captions, suggesting that it makes $\mathbf{u}_{comp}$ more sensitive to the presence of unmatched components than the "average" alternative. However, it shows slightly inferior results on the standard retrieval task. In the experiments, we adopt average pooling as the implementation of semantic aggregation function $\Phi(\cdot)$.

Setting of $\boldsymbol{\alpha}$. One may argue that the combination coefficient α can also be learnable when training the model instead of being a fixed value (0.75 in the experiments). Informally, $\mathbf{u}_{comp}$ imposes a prior that the caption embedding should cover all semantic components in the text. The hyperparameter α controls the strength of this prior (see Figure 8 in the main text for details). Directly learning α under the supervision of the standard retrieval task may encourage the model to focus on only part of the semantic components [4]. Our empirical results support this: α finally converges to 0.93 when treated as a learnable parameter, in contrast to the value of 0.75 suggested in our paper. Shown in Table 4, making α learnable does not affect the performance on the standard retrieval tasks. However, it shows a significant performance drop when there are adversarial captions. Thus, we treat α as a fixed value in UniVSE.

A.3. Hyperparameters

We set the dimension d_{basic} of basic semantic embeddings as 300. The embeddings are initialized by GloVe word embeddings pre-trained on the Common Crawl dataset: http://nlp.stanford.edu/data/glove.840B.300d.zip. The dimension d_{modif} of modifier semantic embeddings is set to 100. The embeddings are randomly initialized. During training, we fix the basic semantic embeddings of words $\mathbf{w}^{(basic)}$. The learning rate of the Adam optimizer is fixed to 0.001 at first 6 epochs and is exponentially decayed by 2 for each next epoch until it reaches 1e-5.

Train Drop	Test Drop	Standard	Obj. Atk.	Attr. Atk.	Rel. Atk.
		469.5	210.9	189.9	**202.2**
✓		468.9	**213.7**	**191.5**	199.4
✓	✓	468.7	211.2	182.2	197.1

Table 1. We evaluate the performance of UniVSE on the standard bidirectional retrieval task and the retrieval tasks with adversarial captions (object-typed, attribute-typed and relation-typed). We use `rsum` as the evaluation metric.

Spatial Aggregation Methods	Standard	Obj. Atk.	Attr. Atk.	Rel. Atk.
Avg	452.9	198.7	184.2	186.2
Max	462.5	209.6	184.1	193.6
Max-10	**469.5**	**210.9**	**189.9**	**202.2**
Max-10 (without ℓ_{comp})	466.1	202.9	182.1	192.2

Table 2. We evaluate the performance of UniVSE under different spatial aggregation settings on the standard retrieval task, and the retrieval tasks with adversarial captions (we report the `rsum`s).

Semantic Aggregation Methods	Avg	Sum	Max	Self-Att.	GRU
Standard (`rsum`)	469.5	471.1	465.8	467.1	**472.0**
Adversarial (`rsum`)	603.0	604.3	**628.4**	599.5	603.7

Table 3. We evaluate the performance of UniVSE under different semantic aggregation settings on the standard retrieval task, and the retrieval tasks with adversarial captions (we report the sum of `rsum`s under three types of attacks).

Model	Standard	Obj. Atk.	Attr. Atk.	Rel. Atk.
Fixed α (0.75)	**469.5**	**210.9**	**189.9**	**202.2**
Learnable α (0.93)	468.1	204.1	182.2	190.4

Table 4. The performance of UniVSE with a learnable or a fixed α on the standard retrieval task, and the retrieval tasks with adversarial captions (we report the `rsum`s).

B. Experiment Details

B.1. Cross-modal Retrieval

Visualizations. We show a set of examples of the image-to-sentence retrieval in Fig. 1 and sentence-to-image retrieval in Fig. 2.

B.2. Retrieval under text-domain adversarial attack

Experiment setup. We use the 1K test split (including 5,000 captions) for generating adversarial attacks. For each caption, we generate five adversarial captions under one type of attack setting. The detailed settings of the three types of adversarial attack are listed below.

1. **Object attack**: We randomly replace / append by an irrelevant noun for both 50% probability. The replacing/appending place is randomly selected in nouns of the caption. For the case of appending extra noun, the word `and` is also added before the appended noun, *e.g.*, `A dog eats meat` →`A dog eats meat` **`and table`**. The irrelevant nouns are drawn from the set containing nouns with high concreteness (manually extracted).

2. **Attribute attack**: If a caption contains attribute-noun pairs. We randomly select one pair and replace the attribute by a negative one. If a caption does not contain any attributes, we randomly choose one noun in the caption and append an attribute on it. The negative attribute is generated from the attribute set excluding the attributes (and its similar attributes) in the caption. The similar attribute group is defined as the following. `{white, snowy, polar}`, `{red, pink}`, `{blue, cloudy}`, `{green, grassy}`, `{brown, sandy, yellow, orange}`, `{rocky, concrete}`.

3. **Relational attack**: For those captions containing relational phrases, we randomly select one relation triple and with equivalent probability to choose one in the triple to be replaced by an irrelevant one. *e.g.*, `A dog eats meat` → `A dog` **`plays`** `meat`. For those captions which do not have any relational phrases, we first randomly select one noun in the caption and regard it as a subject/object with 50% / 50% probability. Then we draw a relational word and an irrelevant noun as the object/subject to form a new fake relation. *e.g.*, `A dog is sleeping` → `A dog` **`in sky`** `is sleeping`.

Baselines. We train the VSE-C according to the setting in [4] with the officially open-sourced code. In the original VSE-C paper, The VSE-C is trained by generating either noun-typed/numeral-typed/relation-typed or all of these three types of adversarial samples. We use the setting of training under all types of adversarial samples as a comparable competitor in this evaluation. For the ablation of UniVSE ($\mathbf{u}_{sent} + \mathbf{u}_{attr}$) (*i.e.*, use $\mathbf{u}_{attr}$ as $\mathbf{u}_{comp}$) under attribute attack scenario, we additionally include $\mathbf{u}_{obj}$ to $\mathbf{u}_{comp}$. The reason is that the attribute attack may add new attribute modifier on a sentence with *no* attributed phrases. $\mathbf{u}_{comp}$ is not defined for such sentence if we only use $\mathbf{u}_{attr}$ as $\mathbf{u}_{comp}$, since it does not contain any attributed phrases. As a solution, we additionally include $\mathbf{u}_{obj}$ to $\mathbf{u}_{comp}$ (*i.e.*, $\mathbf{u}_{comp} = \Phi(\{\mathbf{u}_{attr}\} \cup \{\mathbf{u}_{obj}\})$) to ensure $\mathbf{u}_{comp}$ is well defined even there is no attributed phrases in the sentence.

Visualizations. We show a set of examples of image-to-text retrieval under text-domain adversarial attack in Fig. 3.

B.3. Unified text-to-image retrieval

Experiment setup. We use the 1K test split as the retrieval set. The queries are generated from frequent semantic components extracted by the semantic parser from the training set. We regard a query as a valid one if at least 3 images (5 for noun-level retrieval) in the test set contain the query. For the obj(det) queries, we directly use the class names of the MS-COCO object detection / segmentation annotations.

Baselines. For VSE++ and VSE-C, as they do not have an object-level encoder. For any query, we always regard it as a short sentence and feed it into the sentence encoder to get the embedding of the query text. For UniVSE, as it has the object-level encoder which means a noun/attribute-noun pair can be either encoded by the object encoder ϕ or by neural combiner ψ by regarding the query as a short sentence. We select the encoder having higher performance on a validation set and report the results.

Visualizations. We show a set of retrieved image by queries of various types in Fig. 4.

B.4. Semantic Parsing

Experiment setup. We also use the 1K test split for this experiment. For each caption, we first extract nouns, adjectives and relational words. We call adjective and relational words as content words. The model should recover the dependencies linked with them. We exclude some relational words whose lexical meanings are usually ambiguous, such as `include`, `to`, `of`, *etc*.

Given a content word (either an adjective or a relational word), we generate all possible dependencies among nouns in the sentence to form candidate dependencies. Each candidate dependency, which is either an adjective-noun pair or a subject-relation-object triple, will get a matching score w.r.t. the image (the *visual cue*). We select the dependency that has the highest score as the recovered dependency w.r.t. the chosen content word.

Metrics. We report the accuracy of the recovered semantic dependencies. In detail, for an attribute-noun dependency, the model gets a correct count if the dependency having the highest matching score is identical to the ground-truth. For the dependency of a relation, the model gets 0.5 correct counts if the subject/object of the answer is the same as the ground-truth. If both of them are the same as the ground-truth, the model gets 1 correct count. The reported accuracy computed as the fraction between total correct counts and the total number of dependencies.

Visualizations and failure case study. Shown in Fig. 5, we visualize some successful and failure cases in semantic parsing with visual cues. Error source analysis is also provided.

B.5. Embedding Visualization

We visualize the semantic space of different semantic levels by t-SNE [3]. The result can be found in Fig. B.5. Through the joint learning of vision and language, our unified VSE space successfully recovers the similarities between semantic components at various levels.

References

[1] F. Faghri, D. J. Fleet, J. R. Kiros, and S. Fidler. VSE++: Improving Visual-Semantic Embeddings with Hard Negatives. In *Proceedings of British Machine Vision Conference (BMVC)*, 2018. 1

[2] Z. Lin, M. Feng, C. N. d. Santos, M. Yu, B. Xiang, B. Zhou, and Y. Bengio. A Structured Self-attentive Sentence Embedding. *arXiv:1703.03130*, 2017. 2

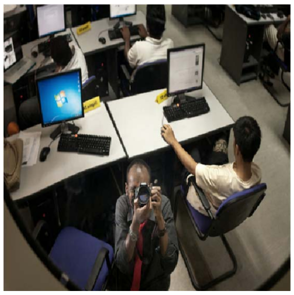
[3] L. v. d. Maaten and G. Hinton. Visualizing Data Using t-SNE. *Journal of Machine Learning Research (JMLR)*, 9(Nov):2579–2605, 2008. 4

[4] H. Shi, J. Mao, T. Xiao, Y. Jiang, and J. Sun. Learning Visually-Grounded Semantics from Contrastive Adversarial Samples. In *Proceedings of International Conference on Computational Linguistics (COLING)*, 2018. 1, 2, 4

[1] (0.476) A few people that are playing tennis on a court.
[2] (0.472) Two people playing a match of tennis on a court.
VSE++ [3] (0.460) Several men playing with a soccer ball in a park.
[4] (0.457) Two young men playinga game of soccer.
[5] (0.455) There is a man running on a field with a soccer ball.

[1] (0.428) There are two soccer teams playing a game on the field.
[2] (0.401) A few people that are playing tennis on a court.
VSE-C [3] (0.395) Several boys on a field playing with a frisbee.
[4] (0.381) A group of people playing soccer in a field.
[5] (0.374) There are people playing a game of tennis.

[1] (0.456) A man carrying a soccer ball down a field.
[2] (0.454) There is a man running on a field with a soccer ball.
U-VSE [3] (0.440) A man that is on a soccer field with a ball.
[4] (0.429) A man kicking a soccer ball while standing on a field.
[5] (0.411) The soccer player is bringing back the ball into play.



[1] (0.520) A bowl with something in it with a banana next to it.
[2] (0.500) A banana sits by two oranges, a bowl and a white plate on a white tray.
VSE++ [3] (0.498) The banana is laying next to an almost empty bowl.
[4] (0.492) A banana and a nearly empty bowl of food resting on top of a table.
[5] (0.467) A white tray with a banana and two tangerines and a plate and bowl.

[1] (0.465) The banana is laying next to an almost empty bowl.
[2] (0.440) A banana and a nearly empty bowl of food resting on top of a table.
VSE-C [3] (0.423) A white tray with a banana and two tangerines and a plate and bowl.
[4] (0.414) A bowl with something in it with a banana next to it.
[5] (0.360) A bowl filled with leftover food sitting next to a banana.

[1] (0.551) A banana and two oranges sit on a tray next to a bowl and a plate.
[2] (0.519) A bowl with something in it with a banana next to it.
U-VSE [3] (0.506) A banana sits by two oranges, a bowl and a white plate on a white tray.
[4] (0.502) A white tray with a banana and two tangerines and a plate and bowl.
[5] (0.498) The banana is laying next to an almost empty bowl.

[1] (0.373) A couple of horses standing in a field.
[2] (0.353) Two giraffes standing in front of each other.
VSE++ [3] (0.346) A big heard of cows walking down a road in a row with green tags on their ears.
[4] (0.345) Sheep that have been sheared standing in a pen.
[5] (0.344) Mythical character with white horse standing on grooved surface.

[1] (0.325) Ten porcelain pieces with floral patterns painted on them.
[2] (0.310) Two horses have feathers on their head.
VSE-C [3] (0.304) Two giraffes standing in front of each other.
[4] (0.303) Horses standing in shallow water in a wooded area.
[5] (0.298) Two dogs lay next to each other on a brown couch.

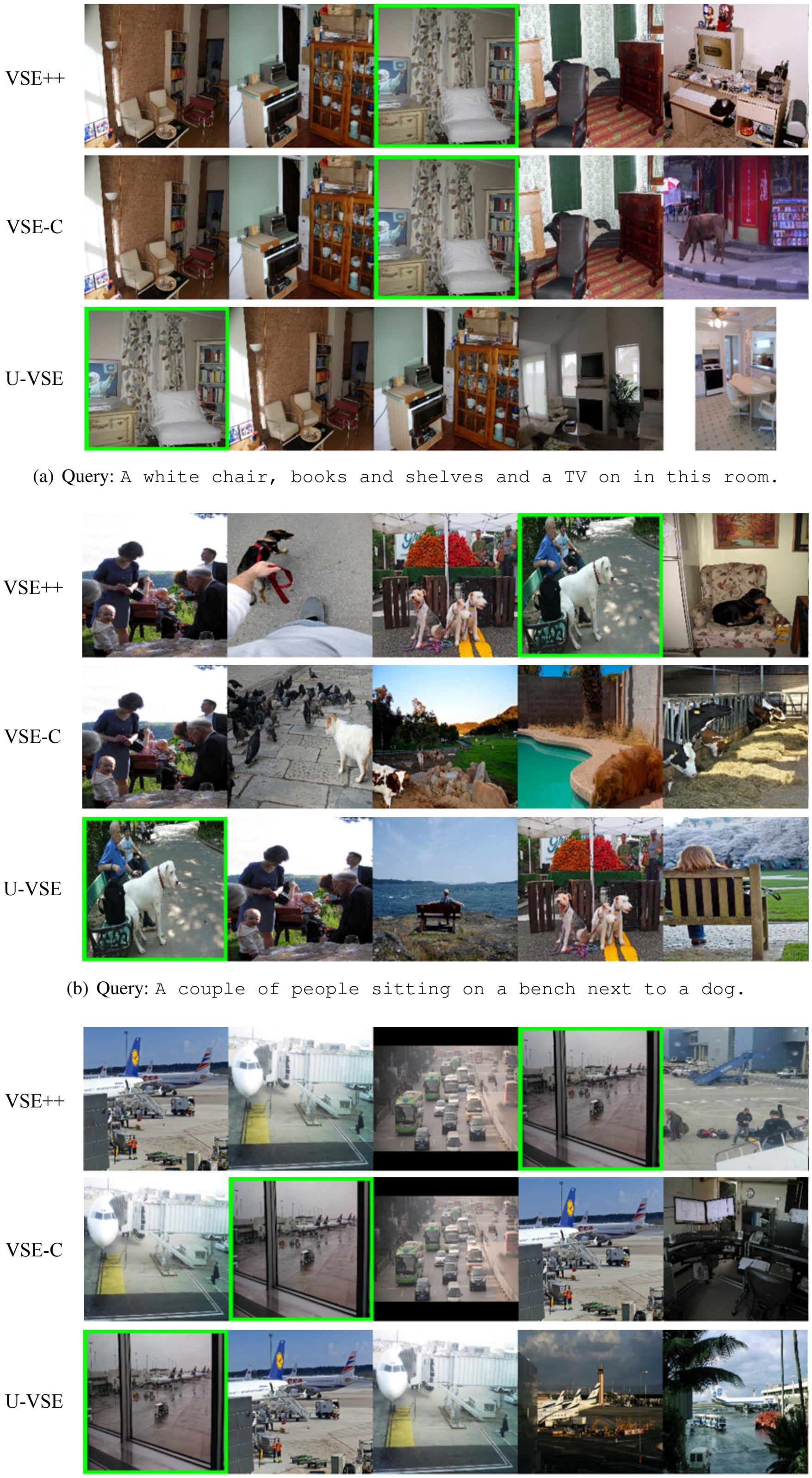
[1] (0.363) Three different horse figurines are placed beside each other.
[2] (0.360) A couple of white horses standing in front of a building.
U-VSE [3] (0.345) Three plastic horse figurines standing next to each other on a shelf.
[4] (0.344) Two horses with red feathers on top of their heads.
[5] (0.339) Three model horses on a table in front of a pegboard backdrop.

[1] (0.410) A basketball player holds a basketball for a picture.
[2] (0.395) A woman standing in the dark holding up a cell phone.
VSE++ [3] (0.383) A person with a basketball stands in front of a goal.
[4] (0.371) A young woman is posing for camera.
[5] (0.352) A woman standing next to another woman in a building.

[1] (0.322) A woman hugging a girl who is holding a suitcase.
[2] (0.297) A young woman is posing for a camera.
VSE-C [3] (0.296) A young man in green jersey is holding a ball.
[4] (0.290) A woman with her arms around a girl who is holding a suitcase.
[5] (0.288) A woman standing in the dark holding up a cell phone.

[1] (0.363) A basketball player holds a basketball for a picture.
[2] (0.360) A uniformed boy is holding a basketball with his back to the hoop.
U-VSE [3] (0.345) A person with a basketball stands in front of a goal.
[4] (0.344) Two basketball players reach up for the hoop.
[5] (0.339) Two basketball players jump to the hoop to block another from scoring.

Figure 1. Examples showing the top-5 image-to-text retrieval results. We highlight the positive captions in blue. The score in the front of each sentence is the similarity score of the caption and image computed by different model. Best viewed in color.

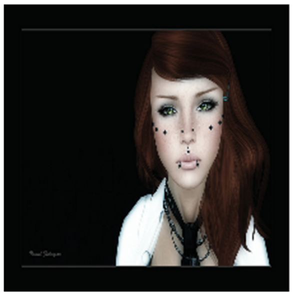


(a) Query: `A white chair, books and shelves and a TV on in this room.`

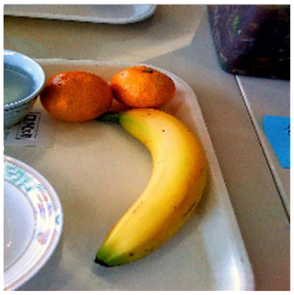
(b) Query: `A couple of people sitting on a bench next to a dog.`

(c) Query: `Window view from the inside of airplanes, baggage carrier and tarmac.`

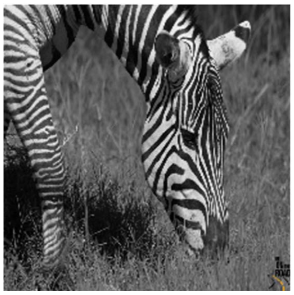
Figure 2. Examples showing the top-5 sentence-to-image retrieval results. We highlight the correct images in green box.



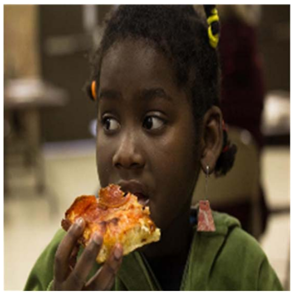
VSE++	[1] (0.481) A woman holding a scissor close to her hair. [2] (0.467) A woman walk a scissor close to her hair. [3] (0.462) An image of a shorts with a qoute at the top. [4] (0.454) A picture of a woman in a good frame. [5] (0.450) A young woman is posing for a camera.
VSE-C	[1] (0.411) A man wearing a mask behind a sbowboarder. (*sbowboarder* is a typo of *snowboarder* in the annotation) [2] (0.400) A man wearing a mask is hold some woodworking. [3] (0.400) A young woman is posing for a camera. [4] (0.385) A man wearing a mask with a sbowboarder. [5] (0.371) A man near a mask with a sbowboarder.
U-VSE	[1] (0.455) A young brunette woman with multiple face piercings. [2] (0.451) A young woman with green eyes and piercings all over her face. [3] (0.434) A young woman is posing for a camera. [4] (0.424) A young woman with green eyes and piercings all stand her face. [5] (0.423) An image of a very cute girl with face piercings.



VSE++	[1] (0.573) Several men sitting at a desk with a computer and stone while another man holds a camera upward. [2] (0.562) Several men sitting at a desk with a computer and while another man holds a camera and sign upward. [3] (0.555) Several men sitting at a desk with a computer while another man holds a camera upward. [4] (0.546) A cellphone of young men sitting at a computer desk. [5] (0.535) Several men sitting at a desk with a computer and while another man holds a camera and pot upward.
VSE-C	[1] (0.462) Several men sitting at a desk with a computer and while another man holds a camera upward. [2] (0.442) Several men sitting at a desk with a computer and while another man holds a camera and sign upward. [3] (0.433) A person talking on a large cell phone and phones. [4] (0.428) A person in glasses is using a laptop and phone. [5] (0.412) People are looking at computer and one man has a camera.
U-VSE	[1] (0.540) People sitting at computers and one person holding a camera. [2] (0.510) Several men sitting at a desk with a computer while another man holds a camera upward. [3] (0.496) People are looking at computer and one man has a camera. [4] (0.490) People sitting at computers and one beds holding a camera. [5] (0.489) A cellphone of young men sitting at a computer desk.



VSE++	[1] (0.577) A black and white photograph of a zebra cat. [2] (0.573) A large group and wall of zebra standing in the grass. [3] (0.566) A black and white photograph of a zebra. [4] (0.550) A large group of zebra standing in the grass. [5] (0.546) There is a black and white image and planes of a zebra eating grass.
VSE-C	[1] (0.550) There is a black and white image of a zebra eating grass. [2] (0.451) A grassy field with various zebras standing next to each other. [3] (0.446) Those zebras may have lost their carrots and they cold be nearby. [4] (0.434) A group of zebras playing and bananas in a field. [5] (0.434) Those zebras may have lost their elephants and they could be nearby.
U-VSE	[1] (0.519) There is a black and white image of a zebra eating grass. [2] (0.514) An antilope is eating grass in between two zebra. [3] (0.514) A black and white photograph of a zebra grazing. [4] (0.513) A close up of a zebra foraging on some grass. [5] (0.499) A black and white photograph of a zebra cat.



VSE++	[1] (0.604) A small boy with a cloudy shirt is eating a sandwich. [2] (0.579) A small boy with a green shirt is eating a sandwich. [3] (0.565) A small boy with a square shirt is eating a sandwich. [4] (0.558) A small boy with a gray shirt is eating a sandwich. [5] (0.550) A small boy with a brown shirt is eating a sandwich.
VSE-C	[1] (0.449) Man in gray shirt eating something that is green. [2] (0.446) A young girl eating a slice of pizza. [3] (0.444) A little girl eating a slice of pizza in a room. [4] (0.442) A dirty girl eating a slice of pizza [5] (0.436) A little girl eating a slice of orange pizza in a room.
U-VSE	[1] (0.512) A young girl with a green jacket eating a piece of pepperoni pizza. [2] (0.510) Small girl in green shirt holding a slice of pizza to her face. [3] (0.505) A young girl eating a slice of pizza. [4] (0.496) A girl takes a gray bite of her pepperoni pizza. [5] (0.494) A dirty girl eating a slice of pizza.

Figure 3. Examples showing the top-5 image-to-sentence retrieval results with the presence of adversarial samples. We highlight the positive captions in blue. Captions with red words are adversarial samples generated from the original captions. Words in red indicates the irrelevant words in the adversarial captions. Best viewed in color.

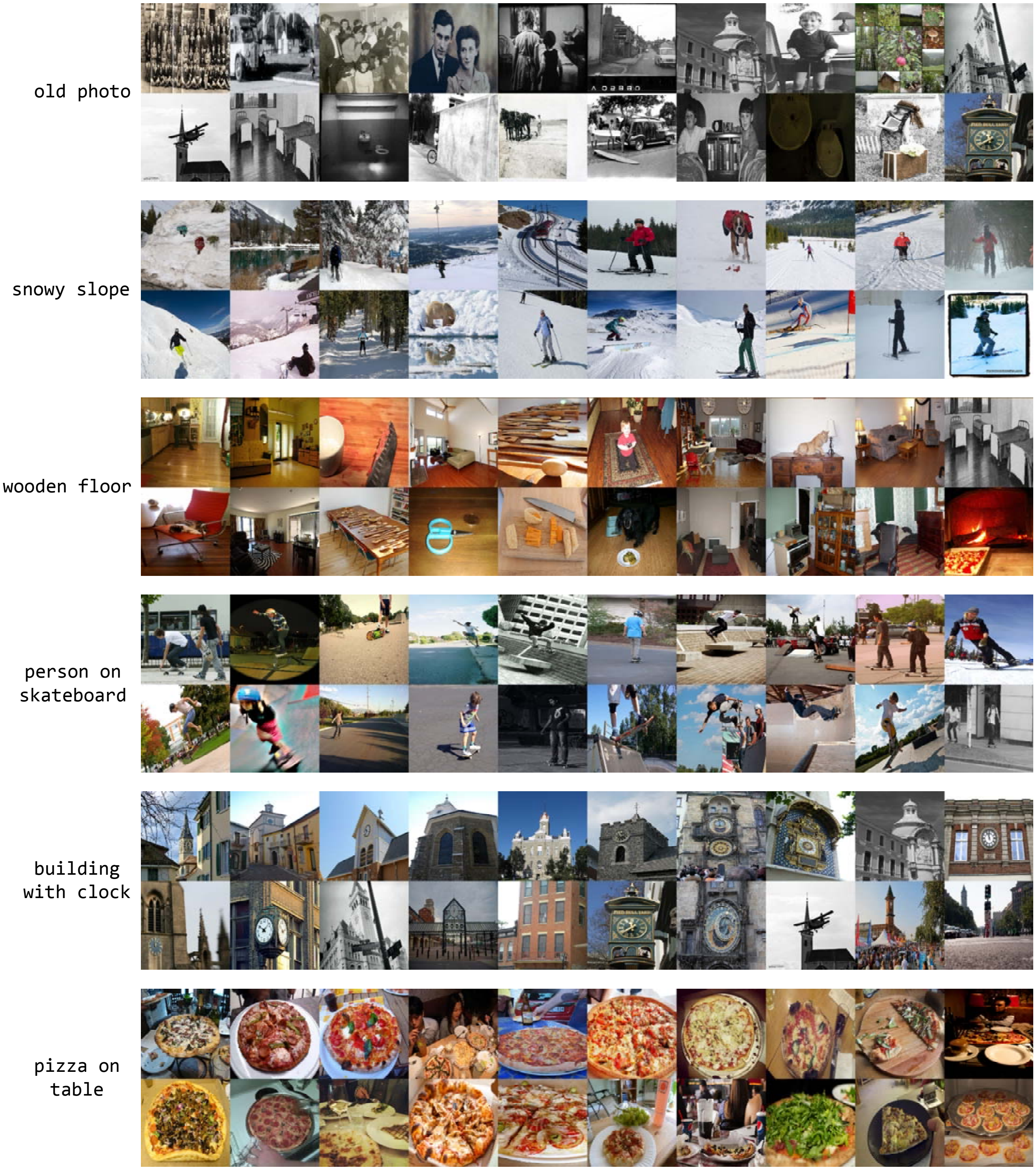


Figure 4. The top-20 retrieved image in the 1K test split set by queries different types: attribute-object pairs and relational triples.

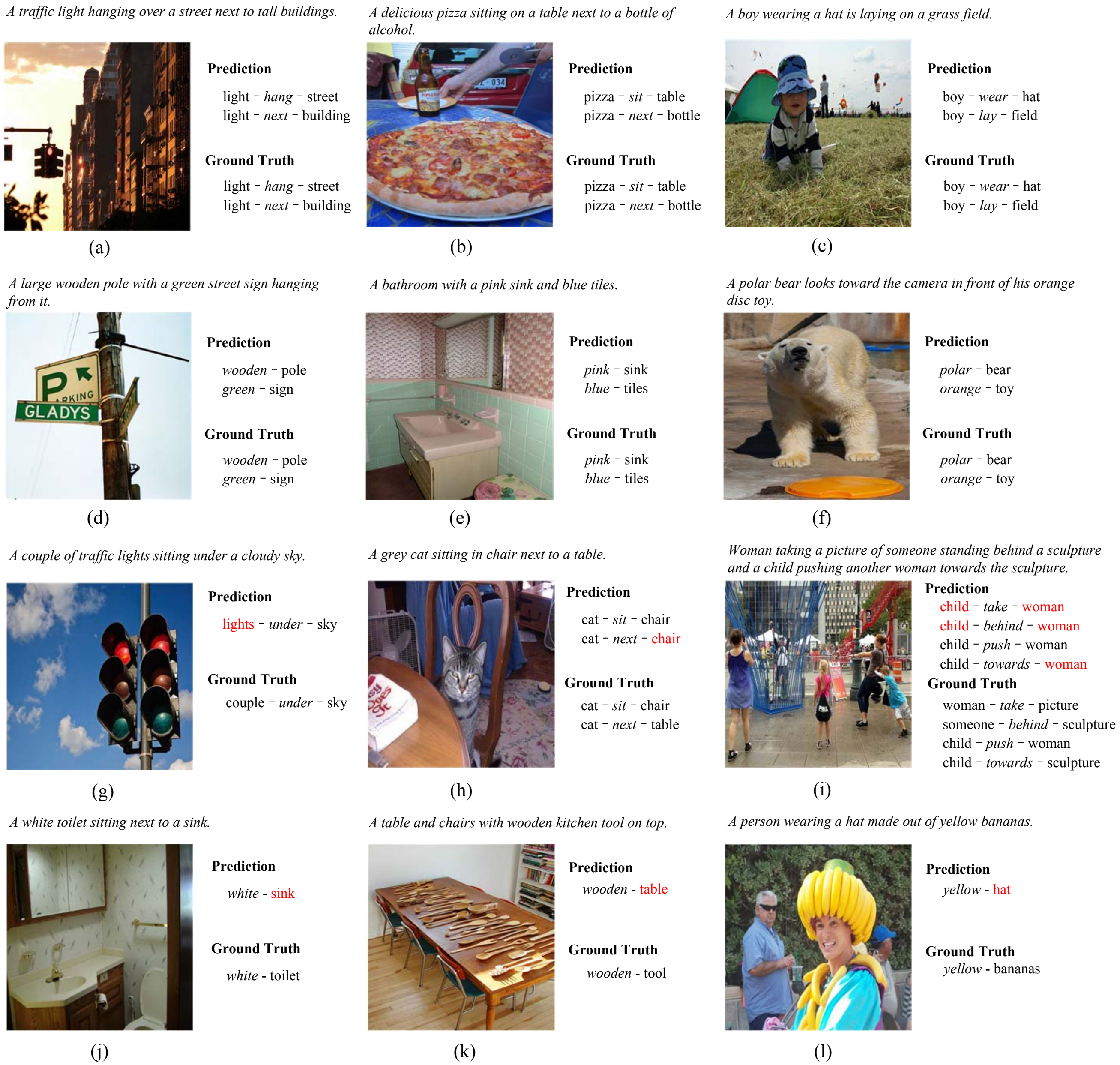
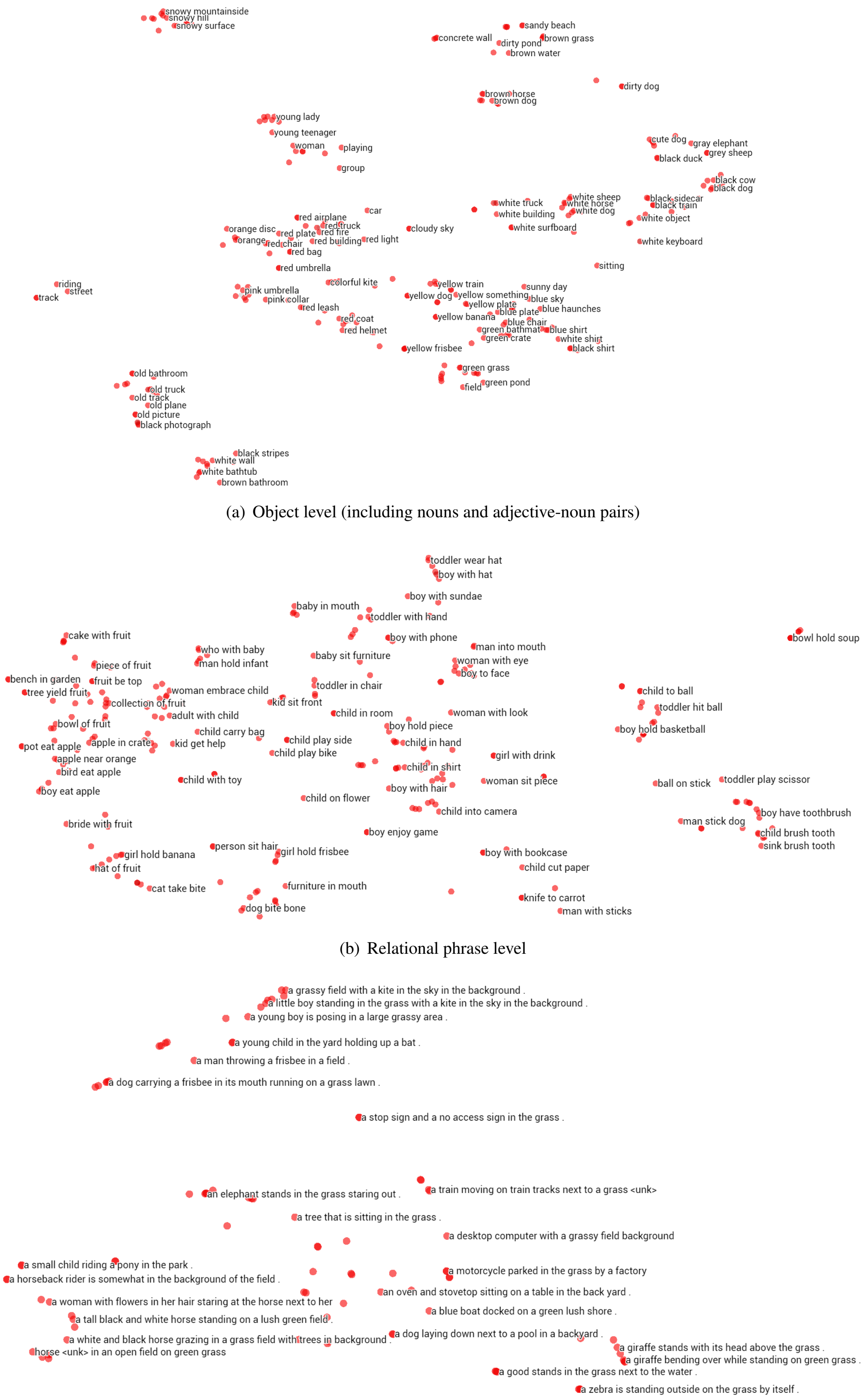


Figure 5. Examples showing the result of semantic parsing based on visual cues. The first and second rows visualize the examples of corrected dependency resolution and the last two rows are the failure cases (dependency resolutions differs from the one by our semantic parser). Words in italic are the content words whose dependency is to be recovered, and the words in red are wrong predictions. Fig. (g) is a failure case of our semantic parser: the word `couple` does not refer to a specific object in the scene. In Fig. (h), both dependencies `cat-next-chair` and `cat-next-table` are actually valid based only on visual cue. Similarly, Fig. (i), (j) and (k) are all cases where only visual cues can not recover the dependency. The result in Fig. (i) shows that our model has the tendency of linking spatially closer objects. In Fig. (l), `hat` and `bananas` actually refers to the same object. Best viewed in color.



(a) Object level (including nouns and adjective-noun pairs)

(b) Relational phrase level

(c) Sentence level

Figure 6. The visualization of the semantic embedding space of different semantic levels. The unified VSE space successfully recovers the similarities between semantic components at various levels.