# Globetrotter: Connecting Languages by Connecting Images

## Appendix

We divide the appendix in two sections. In Section A we show more results, and in Section B we provide more information about the implementation of our method.

## A. Additional results

### A.1. Feature generalization

Training a language model, as opposed to a text representation only designed for image retrieval, has the crucial advantage that it can be finetuned to perform downstream NLP tasks. In this work we are interested in evaluating how well the representations generalize across languages, after training on a downstream task. We evaluate our model on sentence correspondence: we split sentences in two, and half of the times we swap the second half of the sentences with other sentences of the same language. The model has to determine whether or not a sentence is coherent and the beginning of the sentence corresponds to the end of the sentence. We control for uppercase, word breaks, length of sentences etc. so that the model cannot find an easy shortcut (cheat), and has to rely on the semantic and syntactic structure of the sentence. We show examples of the test in Tab. 4 for English.

We train all the models for one epoch on half of the languages in the testing split (first half in alphabetical order), and test on both held-out samples from that half, and *on the languages from the other half* (new languages the sentence correspondence downstream task has not seen). We train a single transformer layer on top of our representation, with one head. For [7], we do not apply the max-pooling over words in order to have a representation for each word. We show results on Tab. 5. The results show that methods trained with language models are much better at performing language tasks. It also shows that our method, trained with alignment, not only performs better on the languages the downstream task has been trained on, but it also generalizes better to other languages the sentence correspondence task has never seen, indicating that the model has a very aligned representation across languages. The relative decrease in accuracy is computed as the percentage decrease of the difference between the accuracy and the chance accuracy.

### A.2. Adaptation to a new language

We test how well our framework can adapt to incoming languages. For this purpose, we test on English and Chinese (separately), which were held out during training. To do so, we precompute features for images and texts from the languages we used during training, and finetune the model for the new language using the same losses as before. We train for one epoch.

After finetuning for English and Chinese, we repeat the same experiments performed for the other languages, showing that our system is able to adapt to new languages without losing the multilingual alignment. See Tab. 1 for translation results, and Tab. 2 for sentence correspondence results. For the sentence correspondence test, we use the head we trained before (without finetuning on the new languages).

### A.3. More results on translation difficulty per language

We show in Fig. 2 the *word* translation accuracy matrix for every pair of languages. As expected, languages that share an important part of their vocabulary are the ones with highest similarity scores. Specifically, there is a very high similarity between Bosnian, Croatian and Serbian, since the three of them are standardized varieties of the Serbo-Croatian language. Also, Indonesian is very close to Malay, as the former is a standardized variety of the latter. A final example is the Czech and Slovak pair: the two of them are languages from the Czech–Slovak group. This shows the importance of cognates across languages. We can find similar patterns for languages that are not as close, but that share the same family or alphabet.

We also show in Fig. 3 the sentence-level translation values we showed in the main paper , but now we plot $A - A^T$. Instead of illustrating which language pairs are close, or are easier to work with, it shows which language pairs are asymmetric in the difficulty of the translation. Rarer languages —e.g. languages that are far from the others in the linguistic tree such as Somali, Tamil or Hindi— are easier to translate from than to translate to.

### A.4. Generated translations

The learned representations are not only good to do translation by retrieval, but also to generate translations. In order to do so, we use a GPT-2 decoder (small version) from [5], pretrained on English. Next, we finetune it on English sentences from our dataset, and after that we finetune it yet again but conditioning it on feature vectors from the English finetuned model from Section A.2. To do this we use an extra linear layer at the input, and we concatenate the results with the input word embeddings. After that, we obtain a GPT-2 model that generates sentences in English based on the input representation. We then test it for translation by inputting representations obtained from other languages, and generating English translations for them. The sentences we used in the test were not used for any of the GPT-2 finetuning stages. We show results in Fig. 4. We selected the

| | English retrieved positives (%) | Chinese retrieved positives (%) |
|---|---|---|
| Chance | 0.48 | 0.48 |
| Text only | 19.27 | 12.98 |
| [7] | 59.18 | 37.96 |
| Globetrotter (Ours) | **75.67** | **62.81** |
| Supervised | 94.87 | 92.77 |

Table 1. Sentence translation results for finetuning. See Appendix A.2.

| | English accuracy (%) | Chinese accuracy (%) |
|---|---|---|
| Chance | 50 | 50 |
| Text only | 65.97 | 55.75 |
| [7] | 50.2 | 50.5 |
| Globetrotter (Ours) | **73.27** | **67.17** |
| Supervised | 69.17 | 62.14 |

Table 2. Sentence correspondence results for finetuning. See Appendix A.2.

first 10 translations that were generated, without any cherry-picking. Interestingly, while our framework is not able to do an accurate literal translation, it does base the translation on the contextual knowledge provided by vision.

### A.5. Comparison with CLIP

As a high-water mark for cross-modal retrieval (in English), we evaluate CLIP [4] on the same cross-modal retrieval regime as in Fig. **??** in the paper, and show results in Table 3. We find that it outperforms our model by around 10-15%, but we note that CLIP has been trained on much more data, exclusively in English, and explicitly for the crossmodal retrieval task. We also attempt to evaluate CLIP in other languages, and naturally find a significant decrease in performance – an order of magnitude worse than our model– though it still outperforms chance (1%).

Note that, by nature, CLIP cannot do machine translation, which is the focus of our work. While learning strong crossmodal matching functions is crucial to our model, it is not the task we aim to solve; we do not attempt to match or outperform CLIP on this task.

| | | CLIP | | Globetrotter (ours) | |
|---|---|---|---|---|---|
| | | $I \rightarrow T$ | $T \rightarrow I$ | $I \rightarrow T$ | $T \rightarrow I$ |
| English | R@1 | 59.55 | 54.57 | 37.33 | 35.40 |
| | R@5 | 82.93 | 79.57 | 71.47 | 68.80 |
| | R@10 | 89.11 | 86.69 | 79.21 | 79.73 |
| All other languages | R@1 | 6.67 | 3.96 | 37.84 | 35.11 |
| | R@5 | 13.98 | 9.01 | 67.56 | 66.19 |
| | R@10 | 18.01 | 12.17 | 77.14 | 76.11 |

Table 3. Cross-modal retrieval results on CLIP. We show Recall@K results for both image to text ($I \rightarrow T$) and text to image ($T \rightarrow I$) directions. All values are percentages. See Section A.5.

### A.6. Clustering in the representation space

In this experiment, we show how differently the representation space is clustered when we train with and without visual alignment. We extract features for the test set examples both for the full model and the text-only model, and cluster these features using k-means, with $k = 50$ clusters. In Fig. 1 we show three sentences belonging to each one of the first three clusters (the selection of both the sentences and the clusters is arbitrary). When training with visual alignment the clusters have a semantic meaning, and when training without it the clusters are language-specific, proving that cross-modal alignment is necessary to obtain good semantic representations.

## B. Implementation details

### B.1. Training and architecture details

We train a transformer network with 4 attention heads and $M = 4$ hidden layers, with a hidden size of $d = 512$. The size of the embeddings at the output of the heads (where the contrastive losses are computed) is $D = 128$. We use a batch size of 800. We set all the $\lambda$ values in the total loss function to $\lambda = 0.2$. We train with an Adam optimizer and a learning rate of $1e-4$.

As mentioned in the architecture section in the main paper , we normalize the feature values $z$ so that $||z||_2 = 1$. Then the similarity value is computed with a dot product,

## Clusters in full model

**Cluster 1: Savannah animals**
(Arabic): یه گورخر که داره به یه گورخر دیگه نگاه میکنه پایین یه مسیر خاکی
(Croatian): popodne provedeno igrajući se sa slonovima
(Georgian): ფართო გასროლა, ჟირაფები სავანას გავლით

**Cluster 2: Wedding**
(Bengali):উইন্ডোতে নববধূ এবং বর
(Slovenian): nevesta v meri obleko, ki ima roza šopek
(Urdu): شخص آپ کی شادی کے دن خواب سچ ہو بنانے دو!

**Cluster 3: Bicycle/Motorcycle**
(Swedish): en cykel kastad ner i sanden på en strand.
(Japanese): 砂地の隣にモーターバイクが駐車しています。
(Tamil): உடற்பயிற்சி பைக் மீது பெண்.
.

## Clusters in text-only model

**Cluster 1: French**
un grand éléphant se tient près d'une clôture
motif circulaire sur fond rouge
homme silhouette à la plage

**Cluster 2: Hindi**
हाथ का एक सेट - डिजाइन के लिए प्यारा फल खींचा.
एक मॉडल घटना के दौरान फैशन शो में रनवे चलता
एक पतली परत पिज्जा तिमाही टुकड़ों में विभाजित।

**Cluster 3: Greek**
ποταμός είναι ένα δημοφιλές σημείο για κανό.
παλιά πόρτα σε ένα ξεχασμένο κήπο
πράσινα ψάρια στο γύρο ενυδρείο.

Figure 1. Clustering in the representation space. When trained without visual alignment the clusters are language-specific, and when trained with visual correspondence the clusters have a semantic meaning.

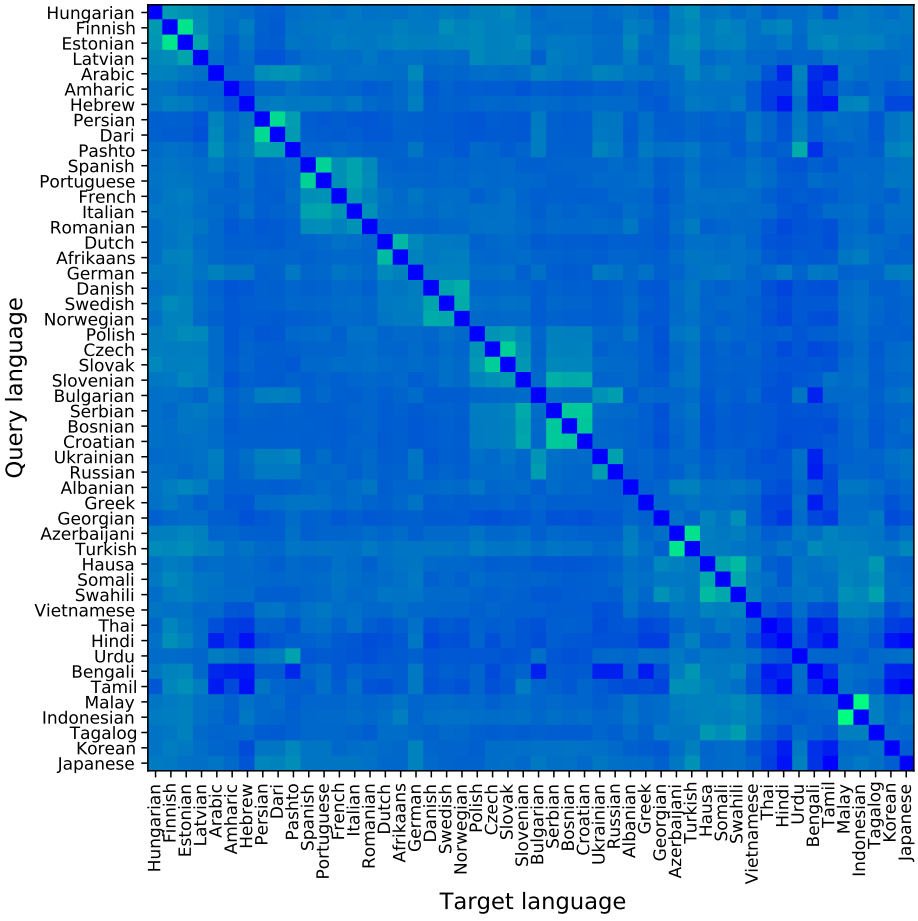


Figure 2. Word-level similarity across languages. See Section A.3 for more information.

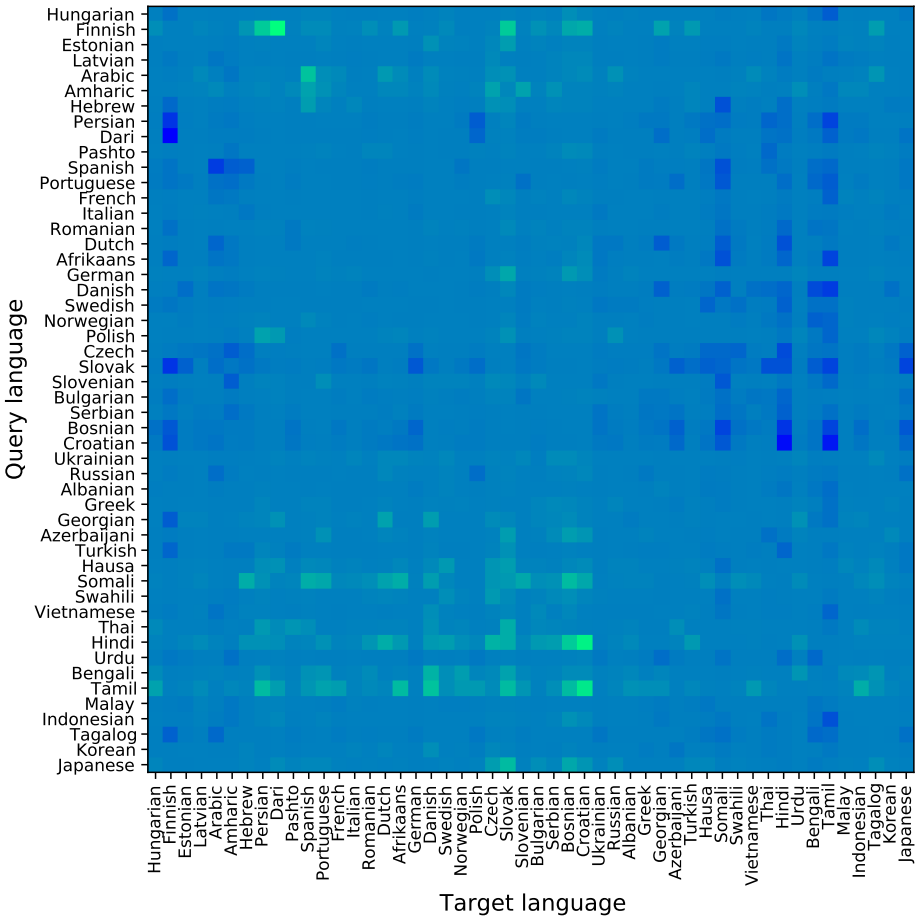


Figure 3. Asymmetry in the direction of the sentence-level translation. See Section A.3.

resulting in the cosine similarity. After that, we scale the value so that the range of the similarity is in $[0, 1]$, instead of $[-1, 1]$.

## B.2. Ground truth for word translation

In order to generate the ground truth translations at the token level, we use the split of the dataset that is translated to all the languages. We then create ground truth token translations for every language pair separately. In order to do that, we follow the tf-idf algorithm. We exploit the fact that we have alignments of languages at the group-of-words (sentence) level. The idea is that if the word "car" appears in an English sentence every time that the word "voiture" (car in French) appears in its French translation, they probably mean the same. In the following explanation, assume we are looking for the translation of a specific token $t_i^A$ from language A into some token $t_j^B$ from language B. We just redefine the concept of "document" in the classical tf-idf algorithm to be the collection of all the words (with repetition) in language B that appear in the same (translated) sentence as $t_i^A$. We call this collection (document) $d$.

First, we create a count of tokens in language B that appear in the document $d$, and compute the *term frequency* (tf) using this count:

$$\text{tf}_{j,d} = \frac{f_{j,d}}{\sum_{j' \in d} f_{j',d}}, \tag{1}$$

where $f_{j,d}$ is the count of the token $t_j^B$ in a document $d$. Second, we compute the inverse document frequency, that takes into account how usual a token is in general, for all $D$ documents:

$$\text{idf}_j = \log \frac{|D|}{|d \in D : t_j^B \in d|}. \tag{2}$$

Multiplying the tf and idf terms we get a value for each $(i, j)$ pairs of tokens (the value is not symmetric). We store tokens $t_i^A$ and $t_j^B$ as ground truth translation if and only if $t_j^B$ is in the top 5 for the tf-idf value of $(i, j)$, for all $j$, *and* $t_i^A$ is in the top 5 for the tf-idf value of $(j, i)$, for all $i$.

| Original sentence | Generated English translation |
|---|---|
| (Russian) кошка отдыхает на обочине в солнечный летний день (cat resting on the curb in sunny summer day) | cat lying on the grass |
| (German) Hardrock-Künstler treten während des Musikfestivals auf (hard rock artists perform during music festival) | artist performs on stage at festival. |
| (Croatian) Nekoliko snowboardera koji su poletjeli niz snijeg prekriveno brdo. (A few snowboarders taking off down a snow covered hill.) | some people skiing in the snow |
| (Arabic) الأوراق النقدية على خلفية سوداء (banknotes on a black background) | silver coin on the black background |
| (German) Porträt auf dem blauen Himmel Hintergrund (portrait on the blue sky background) | bald eagle on the green background |
| (Georgian) დამატებითი ფოტო ქონების ჩამონათვალი (additional photo for property listing) | photo of the front porch |
| (Swedish) utsikt över sjön från rutten (view over lake from the route) | picture of the beach on a sunny day |
| (Hungarian) légi kilátás a strand a legtöbb fehér és tiszta homok (aerial view of the beach with the most white and clean sand) | photo of the mountain lake in winter |
| (Croatian) šetnja po kiši. (a walk in the rain .) | photo of the rain : walking along the streets |
| (Afrikaans) akteur woon die spesiale geleentheid by (actor attends the special event) | some person attends los angeles premiere |

Figure 4. Translation by generation. See Section A.4 for more information.

The following are some examples of translations we obtain between Spanish and English: (electr, electr), (fotograf, ograph), (ción, ction), (grande, lar), (atas, jam), (pare, couple), (decor, decor), (ventana, window), (deportivo, team), (1950, 1950), (form, form), (30, 30), (casa, hom), (lave, key), (1960, 1960), (del, the), (libro, ok), (kara, kara), (ola, surfer), (fan, fan), (viol, viol), (%, %), (dar, standard), (segundo, sec), (equipo, sports), (rojo, red), (árbol, tree), (hierba, gras), (durante, dur), (bron, ze), (mani, demonstr), (pequeño, sm), (tí, typ), (turística, attra), (corre, run), (mus, muse), (atrac, tour), (baño, bat), (mam, mom), (una, on), (element, element), (ijo, son), (ant, ol), (mural, mural), (chocola, chocola), (iste, sad), (cinta, bon), (carro, cart), (edif, bu), (planta, plant), (óc, broccoli), (prim, st), (camina, runway), (cerca, close), (pop, artist), (nacional, nation), (ustr, alian), (vest, dress), (motocic, motorc), (perro, dog), (largo, ong), (+, +), (ates, tom), (fram, rasp), (camina, wal), (inta, inta).

### B.3. Text network details

The input to the text network is a sequence of tokens $\{[SEQ], w_1, \ldots, w_i\}$ that represent a sentence in any language [1]. Before inputting tokens to the transformer, we encode them with a fixed-length vector representation. To embed input tokens, we use a $\mathcal{V} \times d$ word embedding matrix $\phi_w$, where $\mathcal{V}$ is the size of the vocabulary considered by the tokenizer. We use $\mathcal{V} = 30,000$. We augment the input encoding with positional information (word index), translating the encoding by a learned vector: $\phi_{\text{txt}}(w_i) = \phi_w^T w_i + \phi_{\text{pos}}(w_i)$ where $\phi_{\text{pos}}$ encodes the word position of $w_i$.

We then input the augmented tokens to the transformer. A transformer block [8] consists of a multi-headed self-attention layer followed by a linear layer, that outputs a hidden representation for every token in the input sequence. These transformer blocks are concatenated in series to get deeper representations. Let $H^m \in \mathbb{R}^{d \times j}$ be the $d$ dimensional hidden vectors at layer $m$. The transformer first computes vectors for queries $Q = W_q^m H^m$, keys $K = W_k^m H^m$, and values $V = W_v^t H^m$ where each $W_* \in \mathbb{R}^{d \times d}$ is a matrix of learned parameters. Using these queries, keys, and values, the transformer computes the next layer representation by attending to all elements in the previous layer:

$$H^{m+1} = SV \quad \text{where} \quad S = \text{softmax}\left(\frac{QK^T}{\sqrt{d}}\right). \tag{3}$$

In practice, the transformer uses multi-head attention, which repeats Equation 3 once for each head, and concatenates the results. The network produces a final representation $\{h_{[SEQ]}^M, h_1^M \ldots, h_i^M\}$ for a stack of $M$ transformer blocks.

As mentioned in the architecture section in the main paper , we also add a prediction head. This head takes as input the final hidden representation for the $[SEQ]$ token, $h_{[SEQ]}^M$.

### B.4. Dataset details

To collect the dataset, we used captions from the Flickr30k [9], MSCOCO [3] and Conceptual Captions [6] datasets. Flickr30k and MSCOCO are image captioning datasets that have been carefully curated and annotated in a controlled setting, so the text descriptions are accurate and thorough. However, most of the images in our datasets come from Conceptual Captions, which consists of captions harvested from the web, so the visual-language alignment is more noisy.

The list of 52 languages in our dataset is Afrikaans, Albanian, Amharic, Arabic, Azerbaijani, Bengali, Bosnian, Bulgarian, Chinese, Croatian, Czech, Danish, Dari, Dutch, English, Estonian, Finnish, French, Georgian, German, Greek, Hausa, Hebrew, Hindi, Hungarian, Indoniesian, Italian, Japanese, Korean, Latvian, Malay, Norwegian, Persian, Pashto, Polish, Portuguese, Romanian, Russian, Serbian, Slovak, Slovenian, Somali, Spanish, Swahili, Swedish, Tagalog, Tamil, Thai, Turkish, Ukrainian, Urdu, Vietnamese. We further attain ground truth human translations for a subset of the data in the following 11 languages: Dutch, French, Hebrew, Hindi, Italian, Korean, Polish, Portuguese, Russian, Spanish, Turkish.

We randomly split each dataset into 52 equally sized parts, one for each language supported by the machine translation service we use. Each split is assigned a unique language, and splits with the same language across datasets are combined. The split which is assigned the English language is set aside and translated into all 51 other languages, and only used in testing. We also set aside the split translated into Chinese for fine tuning experiments. The remaining 50 splits have their original English captions discarded, and are then split 80%-20% into training and validation data. All experiments shown in the experiments section in the main paper are run on the reserved test data.

Note that there is no overlap at all (visual or linguistic) between the different splits, except for the test split. Please see Table 6 for more details about the dataset.

Finally, in Fig. 5 we show examples of image-caption pairs from the dataset, along with their English translation. This is the same as Figure 4 in the main paper, but adding English translations.

## References


[1] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 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, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. 5

[2] Guillaume Lample and Alexis Conneau. Cross-lingual language model pretraining. *Advances in Neural Information Processing Systems (NeurIPS)*, 2019. 7

[3] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014. 6

[4] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, pages 8748–8763. PMLR, 2021. 2

[5] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. *OpenAI blog 1.8*, 2019. 1

[6] Piyush Sharma, Nan Ding, Sebastian Goodman, and Radu Soricut. Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In *Proceedings of ACL*, 2018. 6

[7] Gunnar A. Sigurdsson, Jean-Baptiste Alayrac, Aida Nematzadeh, Lucas Smaira, Mateusz Malinowski, João Carreira, Phil Blunsom, and Andrew Zisserman. Visual grounding in video for unsupervised word translation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2020. 1, 2, 7

[8] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017. 5

[9] Peter Young, Alice Lai, Micah Hodosh, and Julia Hockenmaier. From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions. *Transactions of the Association for Computational Linguistics*, 2:67–78, 2014. 6

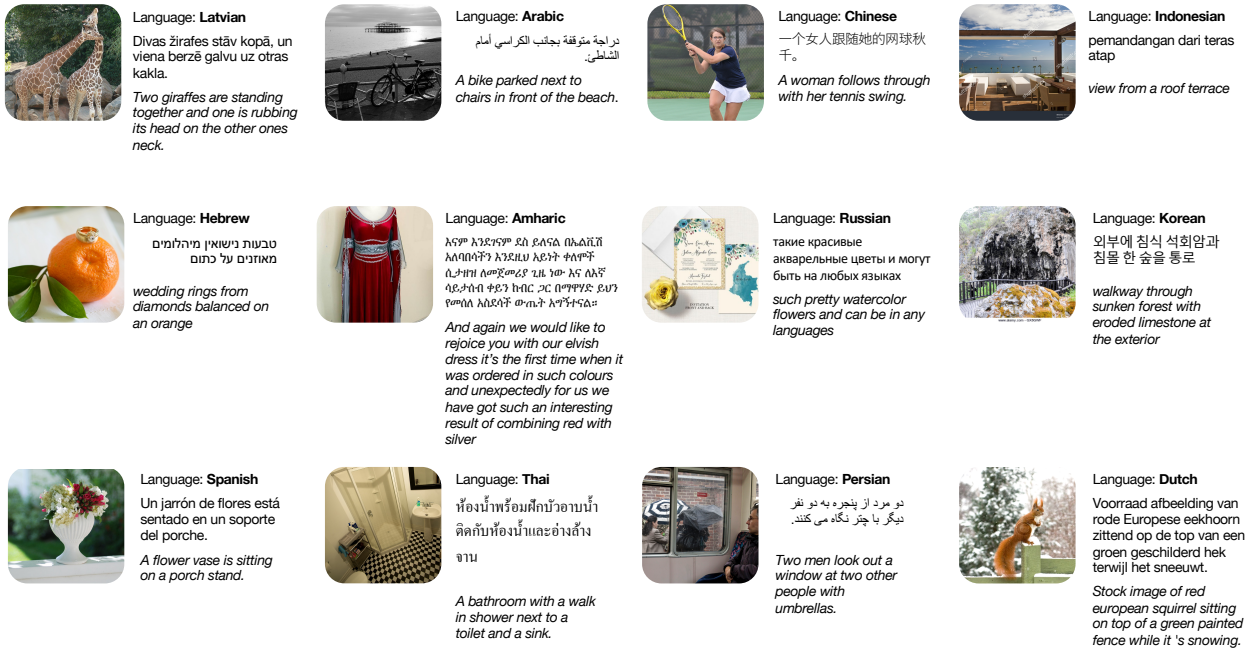


Figure 5. We show some examples of our dataset, along with English translations. Note that we never use the English translations in our framework.

| **Sentence** | **Corresponds** |
|---|---|
| *A piece of cake sitting next to pastries on a white plate with red and yellow sauce* | Yes |
| *Seamless pattern with white bugs on a black background* | Yes |
| *A big tower with a big tv genre and a common language* | No |
| *A hand holding a smartphone with of a picnic by a lake* | No |

Table 4. Sentence correspondence task examples. See Appendix A.1.

| | **Seen accuracy (%)** | **Unseen accuracy (%)** | **Relative decrease (%)** |
|---|---|---|---|
| Chance | 50 | 50 | - |
| Text only | 71.54 | 64.94 | 30.64 |
| [2] | 72.41 | 68.22 | 18.70 |
| [7] | 53.25 | 52.89 | 11.07 |
| Globetrotter (Ours) | **75.95** | **74.54** | **5.43** |
| Supervised | 75.64 | 68.73 | 26.95 |

Table 5. Sentence correspondence results. See Appendix A.1.

| | **Flickr30k** | **MSCOCO** | **Conceptual Captions** | **Total** |
|---|---|---|---|---|
| Image/language pairs per language | $3.1k$ | $11.9k$ | $63.8k$ | $78.7k$ |
| Total image/language pairs | $159k$ | $616k$ | $3.3M$ | $4.1M$ |

Table 6. Dataset statistics. There are a total of 52 languages.