An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale

Alexey Dosovitskiy et. al
ICLR 2021

https://arxiv.org/abs/2010.11929

Contributions

- Transformers in computer vision: applying attention to image patches
- When pretrained on large amounts of data and transferred to multiple mid- or small- image recognition benchmarks, Vision Transformer (ViT) achieves results comparable to SotA convolutional neural networks while requiring fewer resources to train

Motivation

- Self-attention-based architectures, i.e., Transformers have been very popular in NLP
- Dominant approach is to pre-train on a large text corpus then fine-tune on a smaller task-specific dataset
- Transformers are also computationally efficient and scalable (billions of parameters, larger data), no sign of saturating performance

Motivation

- In computer vision, convolutional architectures (CNNs) remain dominant
- Multiple works try to combine CNN with self-attention, but haven’t scaled effectively
- In large-scale image recognition, classic ResNet-like architectures are still state-of-the-art

This Work

- Experiment with applying standard Transformer directly to images with *fewest* possible modifications:
  - Split an image into patches
  - Provide the sequence of linear embeddings of these patches as an input to a Transformer
  - Key: ***image patches*** as ***tokens*** (words) in NLP applications

Findings

- When trained on mid-sized datasets (ImageNet), Transformers yield modest accuracies (a few % below ResNet of comparable size) – expected since Transformers lack inductive biases that CNNs have (e.g., locality) thus do not generalize well when trained on small amount of data
- But, if trained on larger datasets (14M-300M images), large scale training trumps inductive bias – beat SotA on multiple benchmarks

Model

<u>Advantage:</u> can use efficient and scalable Transformers architecture from NLP out of the box!

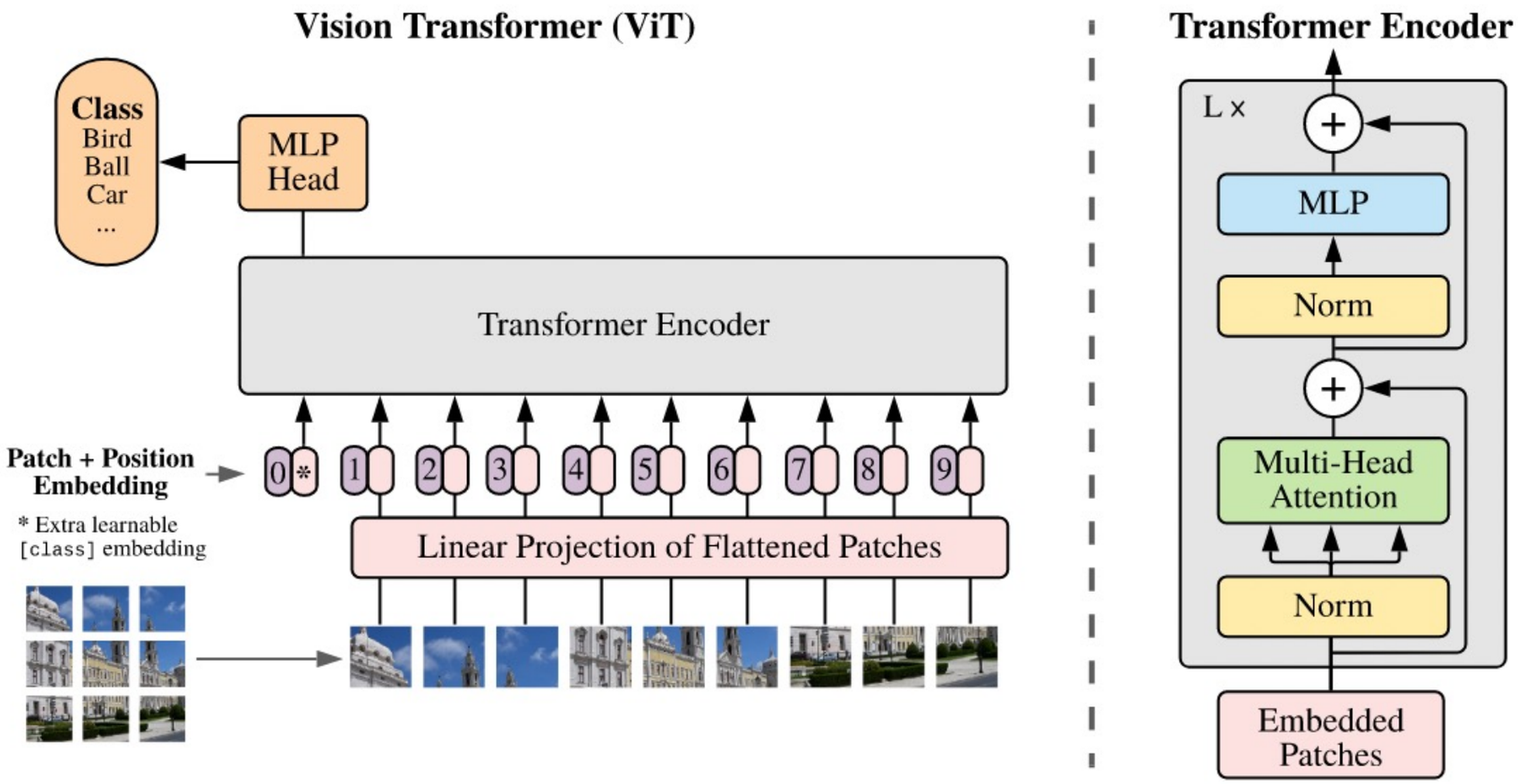


Figure 1: Model overview. We split an image into fixed-size patches, linearly embed each of them, add position embeddings, and feed the resulting sequence of vectors to a standard Transformer encoder. In order to perform classification, we use the standard approach of adding an extra learnable "classification token" to the sequence. The illustration of the Transformer encoder was inspired by Vaswani et al. (2017).

Model

Advantage: can use efficient and scalable Transformers architecture from NLP out of the box!

Model	Layers	Hidden size D	MLP size	Heads	Params
ViT-Base	12	768	3072	12	86M
ViT-Large	24	1024	4096	16	307M
ViT-Huge	32	1280	5120	16	632M

Table 1: Details of Vision Transformer model variants.

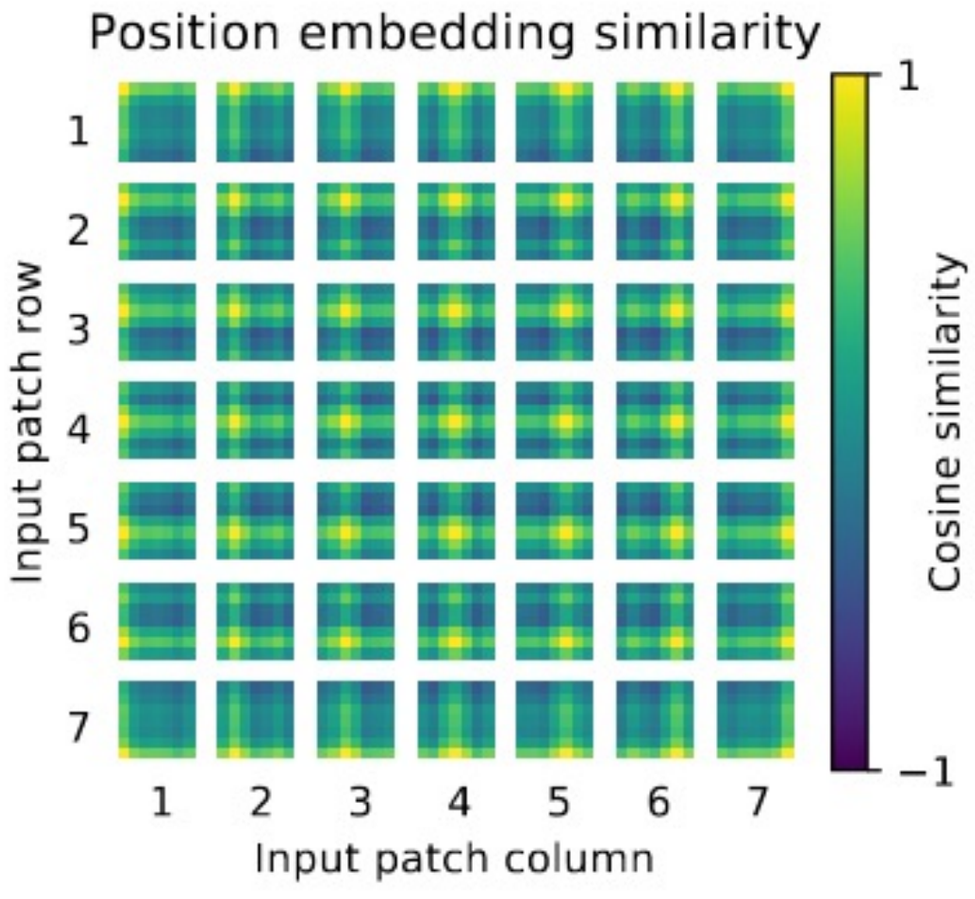
Model

Advantage: can use efficient and scalable Transformers architecture from NLP out of the box!

	Ours-JFT (ViT-H/14)	Ours-JFT (ViT-L/16)	Ours-I21k (ViT-L/16)	BiT-L (ResNet152x4)	Noisy Student (EfficientNet-L2)
ImageNet	**88.55** ± 0.04	87.76 ± 0.03	85.30 ± 0.02	87.54 ± 0.02	88.4/88.5*
ImageNet ReaL	**90.72** ± 0.05	90.54 ± 0.03	88.62 ± 0.05	90.54	90.55
CIFAR-10	**99.50** ± 0.06	99.42 ± 0.03	99.15 ± 0.03	99.37 ± 0.06	–
CIFAR-100	**94.55** ± 0.04	93.90 ± 0.05	93.25 ± 0.05	93.51 ± 0.08	–
Oxford-IIIT Pets	**97.56** ± 0.03	97.32 ± 0.11	94.67 ± 0.15	96.62 ± 0.23	–
Oxford Flowers-102	99.68 ± 0.02	**99.74** ± 0.00	99.61 ± 0.02	99.63 ± 0.03	–
VTAB (19 tasks)	**77.63** ± 0.23	76.28 ± 0.46	72.72 ± 0.21	76.29 ± 1.70	–
TPUv3-core-days	2.5k	0.68k	0.23k	9.9k	12.3k

Table 2: Comparison with state of the art on popular image classification benchmarks. We report mean and standard deviation of the accuracies, averaged over three fine-tuning runs. Vision Transformer models pre-trained on the JFT-300M dataset outperform ResNet-based baselines on all datasets, while taking substantially less computational resources to pre-train. ViT pre-trained on the smaller public ImageNet-21k dataset performs well too. *Slightly improved 88.5% result reported in Touvron et al. (2020).

Model


Position embedding similarity
Input patch row
Input patch column
Cosine similarity
1
2
3
4
5
6
7
1
−1

Learning Transferable Visual Models From Natural Language Supervision

CLIP: Contrastive Language-Image Pretraining

Alec Radford et. Al

https://openai.com/blog/clip/

https://arxiv.org/abs/2103.00020

Caveat: training takes 30 days across 592 V100 GPUs (if you run it on AWS on-demand, you will need to fork out $1,000,000!)

Contributions

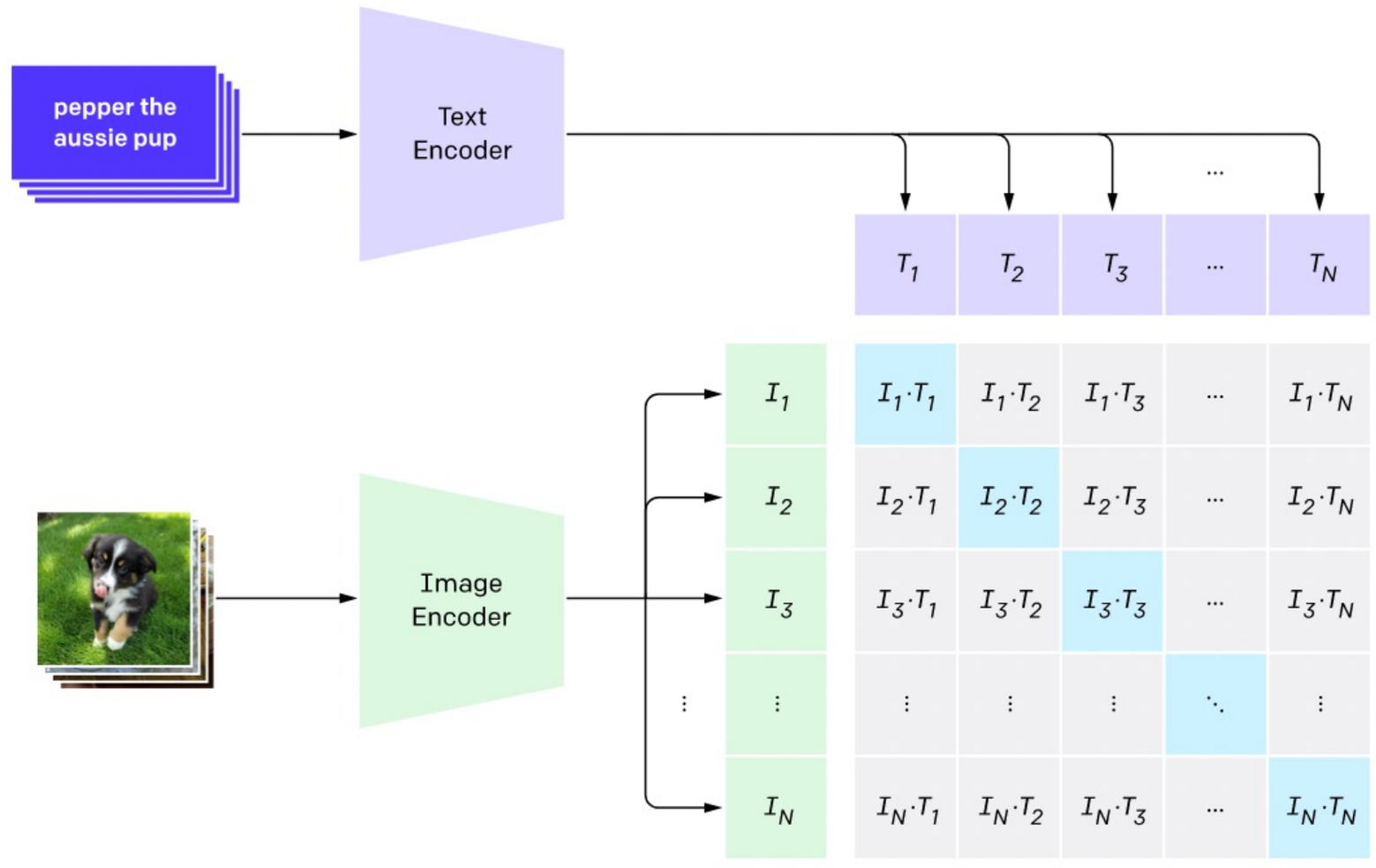
- A vision-language neural network model trained on 400M image-text pairs (pictures-captions in the web)
- Training is to predict the most relevant text snippet given an image
- Can perform well on tasks it is not directly trained to do (zero-shot learning)

Model

- Text encoder: for embedding text
- Image encoder: for embedding image
- Do contrastive pretraining to learn good embeddings for these

Model

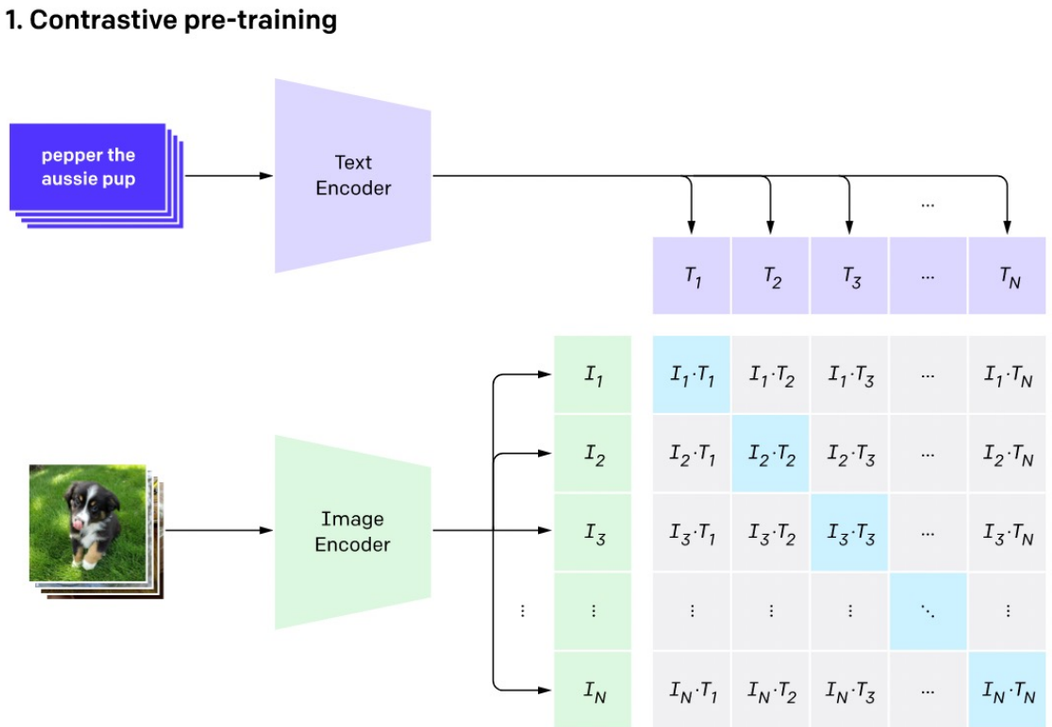
Contrastive pre-training: an embedding for a text should be close to the embedding of an image it is a caption of (cosine-similarity)



Diagonal entries should have high cosine similarities

Non-diagonal entries should have low cosine similarities

Model



<u>Contrastive pre-training:</u> an embedding for a text should be closer to the embedding of an image it is a caption of (cosine-similarity) and further from other images

```
# image_encoder - ResNet or Vision Transformer
# text_encoder  - CBOW or Text Transformer
# I[n, h, w, c] - minibatch of aligned images
# T[n, l]       - minibatch of aligned texts
# W_i[d_i, d_e] - learned proj of image to embed
# W_t[d_t, d_e] - learned proj of text to embed
# t             - learned temperature parameter

# extract feature representations of each modality
I_f = image_encoder(I) #[n, d_i]
T_f = text_encoder(T)  #[n, d_t]

# joint multimodal embedding [n, d_e]
I_e = l2_normalize(np.dot(I_f, W_i), axis=1)
T_e = l2_normalize(np.dot(T_f, W_t), axis=1)

# scaled pairwise cosine similarities [n, n]
logits = np.dot(I_e, T_e.T) * np.exp(t)

# symmetric loss function
labels = np.arange(n)
loss_i = cross_entropy_loss(logits, labels, axis=0)
loss_t = cross_entropy_loss(logits, labels, axis=1)
loss   = (loss_i + loss_t)/2
```

Figure 3. Numpy-like pseudocode for the core of an implementation of CLIP.

Model

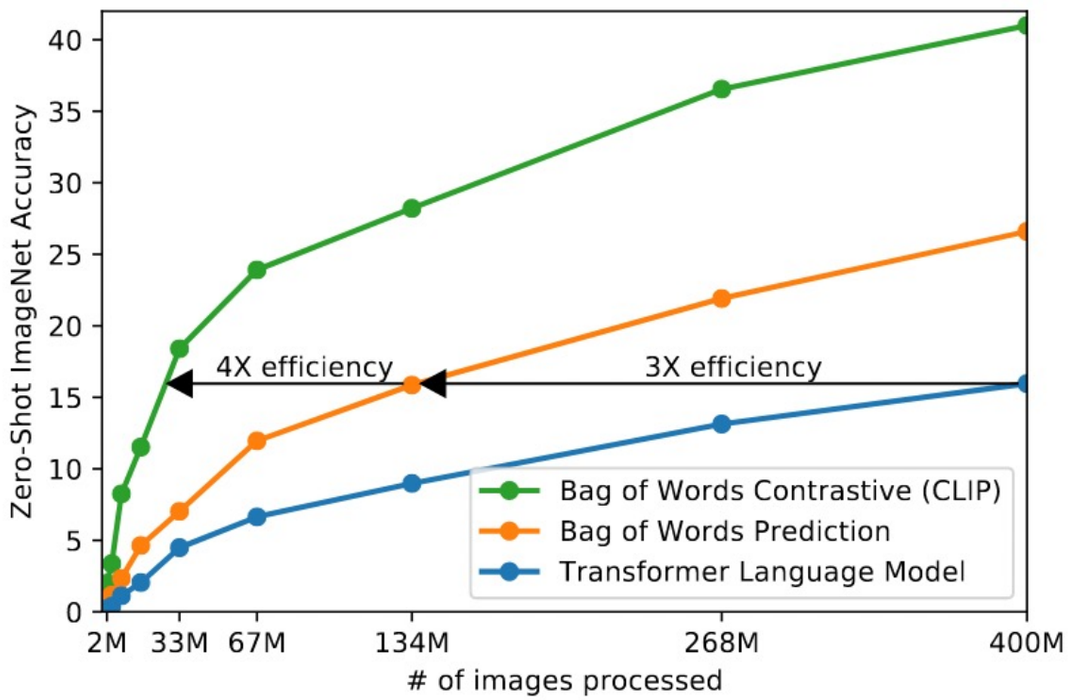


Figure 2. **CLIP is much more efficient at zero-shot transfer than our image caption baseline.** Although highly expressive, we found that transformer-based language models are relatively weak at zero-shot ImageNet classification. Here, we see that it learns 3x slower than a baseline which predicts a bag-of-words (BoW) encoding of the text (Joulin et al., 2016). Swapping the prediction objective for the contrastive objective of CLIP further improves efficiency another 4x.

CLIP: trained to predict which caption is paired with which image (contrastive objective)

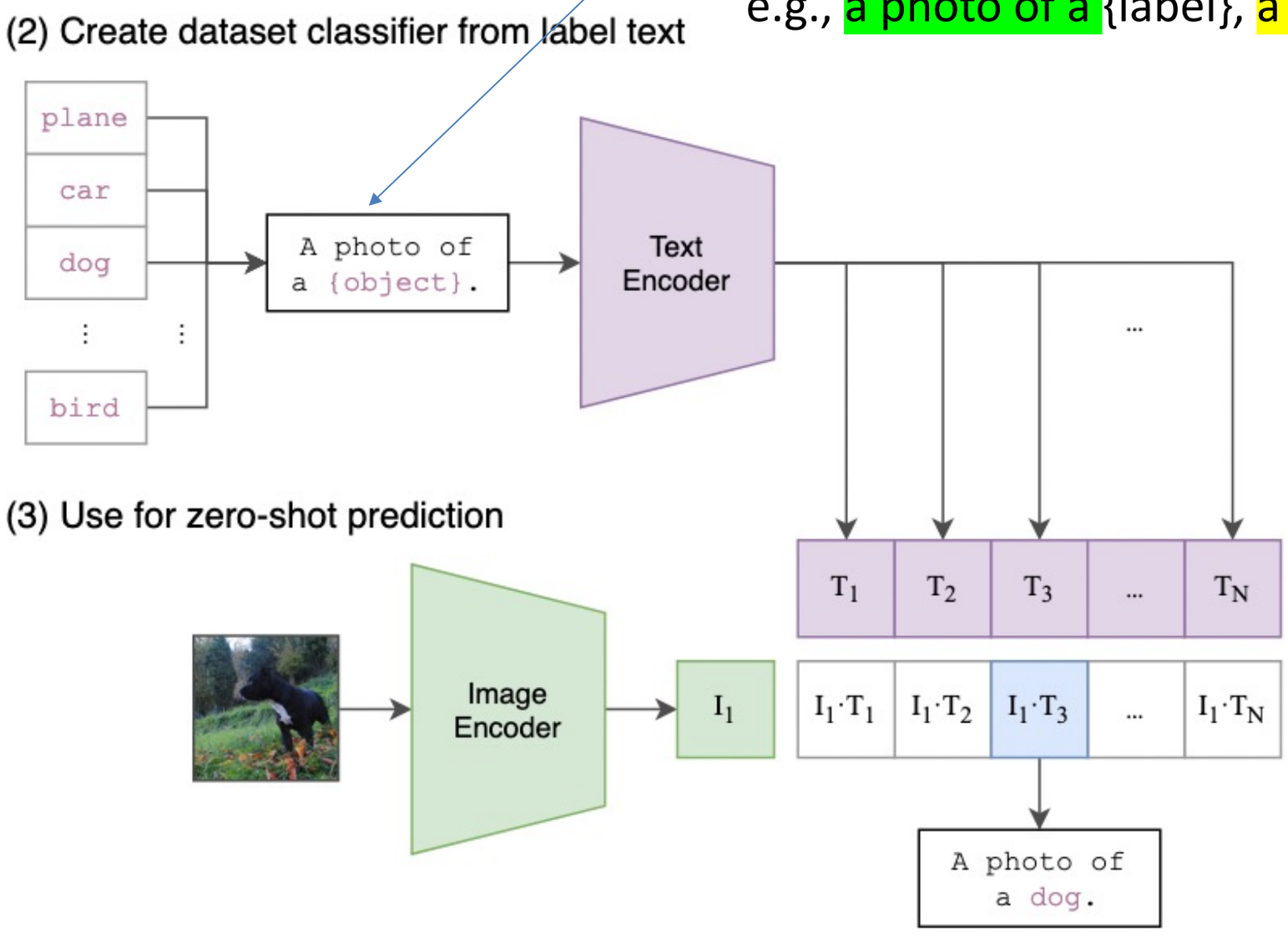
Bag of Words prediction: trained to predict a bag-of-words caption of the image (predictive objective)

Transformer LM: trained to predict caption of an image (predictive objective)

Trained to predict the *exact* words of the text accompanying the image
Harder task

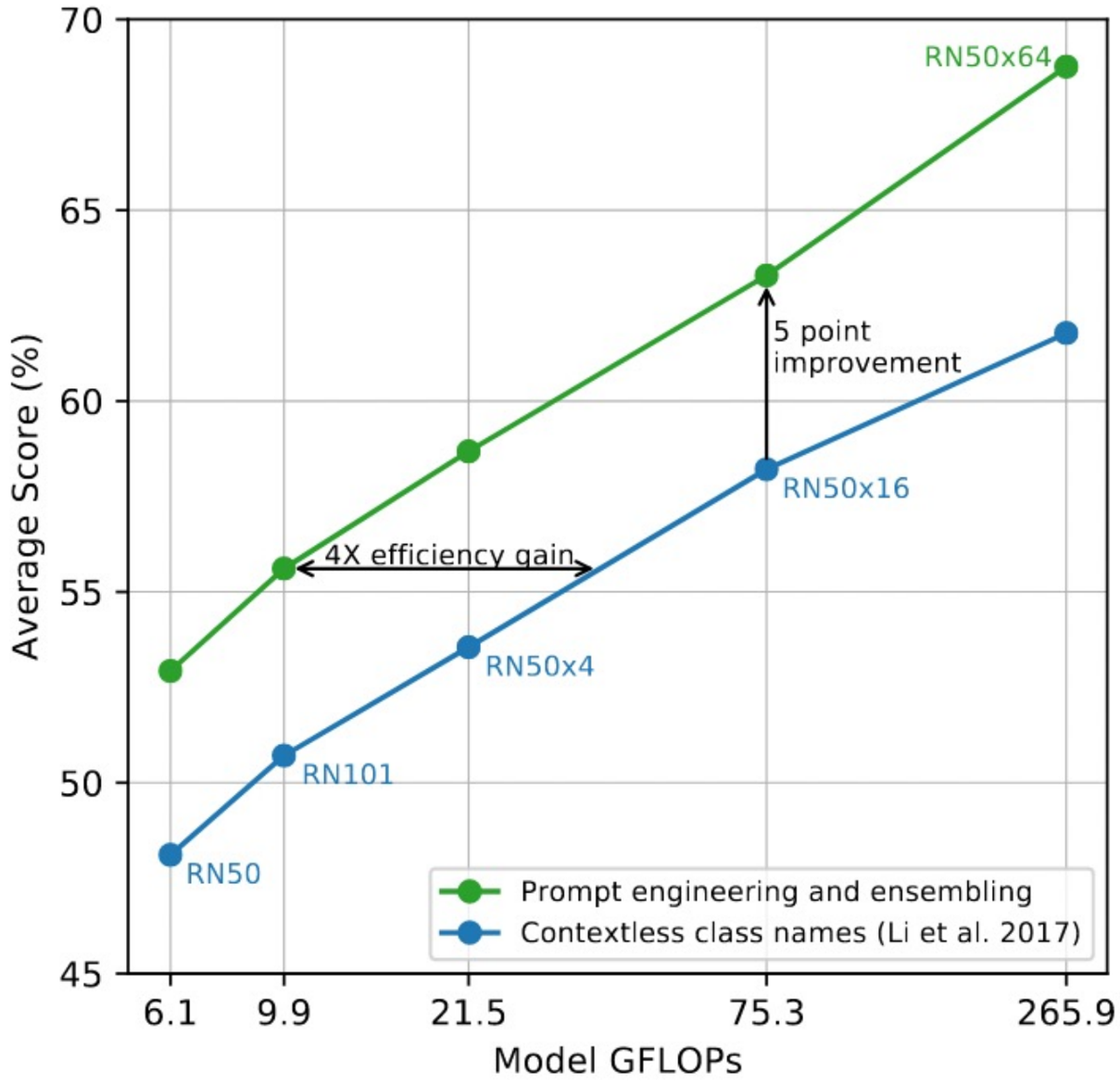
Model

need “prompts”
-- a sentence, to bridge gap between training and test
-- can be customized to each task
e.g., a photo of a {label}, a type of pet



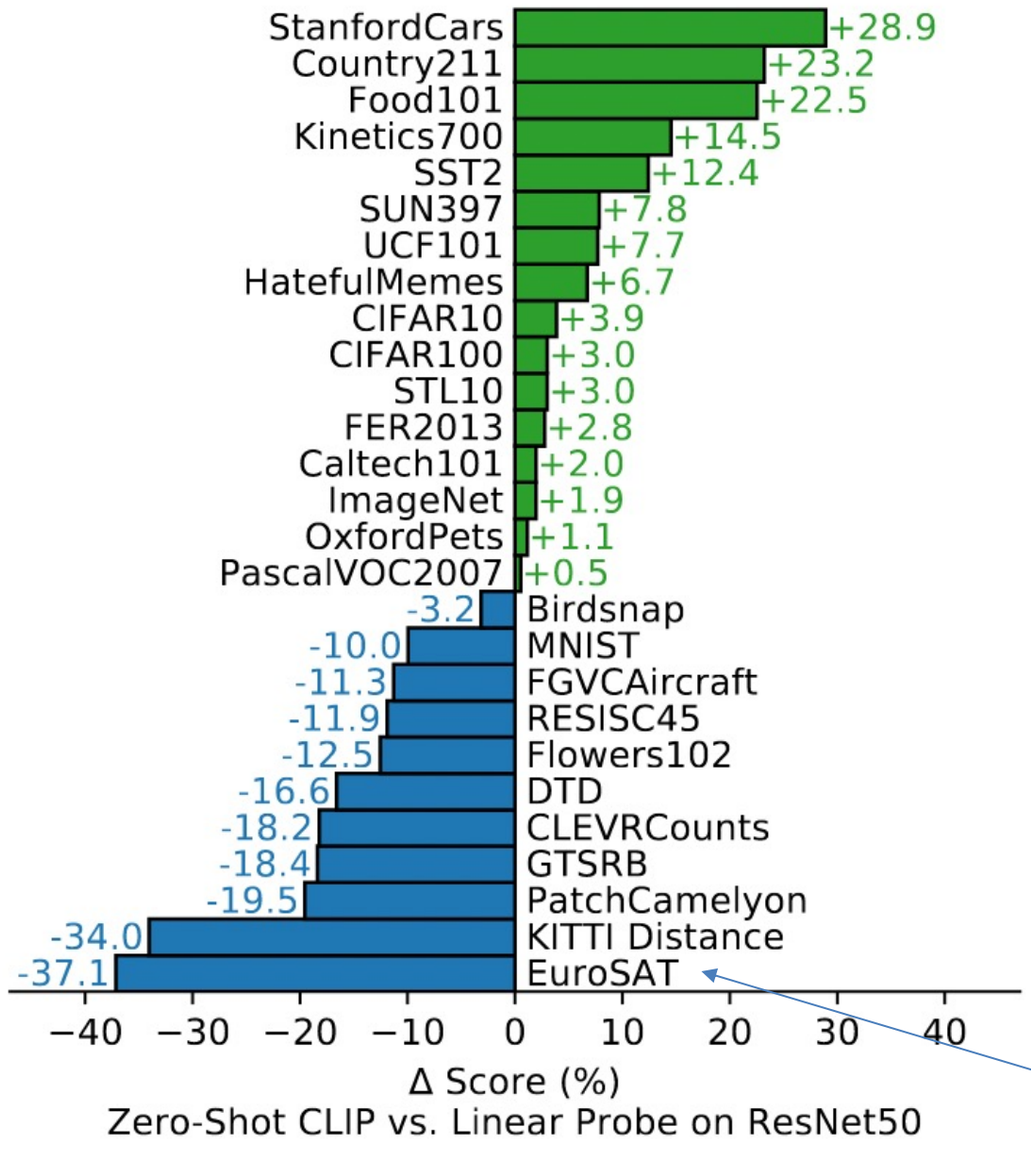
Provide context for the task, also help for polysemy

Prompts



need “prompts”
-- **engineering**: on satellite images, use “a satellite photo of a {label}”; on OCR, use quotes around the number/text to be recognized
-- **ensembling**: ensemble over multiple context prompts: “a photo of a big {label}”, “a photo of a small {label}

Zero-shot performance



Zero-shot CLIP is competitive with a **fully supervised baseline**. Across a **27 dataset** eval suite, a zero-shot CLIP classifier outperforms a fully supervised linear classifier fitted on ResNet-50 features on **16 datasets**, including ImageNet.

Linear Probe on ResNet50: add a linear classification layer (logistic regression) on top of ResNet50, fine-tuned on the target task in a supervised manner

Specialized, complex, or abstract tasks: satellite image classification, lymph node tumor detection, counting objects, ...

Robustness

	Dataset Examples	ImageNet ResNet101	Zero-Shot CLIP	Δ Score
ImageNet		**76.2**	**76.2**	0%
ImageNetV2		64.3	**70.1**	+5.8%
ImageNet-R		37.7	**88.9**	+51.2%
ObjectNet		32.6	**72.3**	+39.7%
ImageNet Sketch		25.2	**60.2**	+35.0%
ImageNet-A		2.7	**77.1**	+74.4%

Zero-shot vs Few-shot

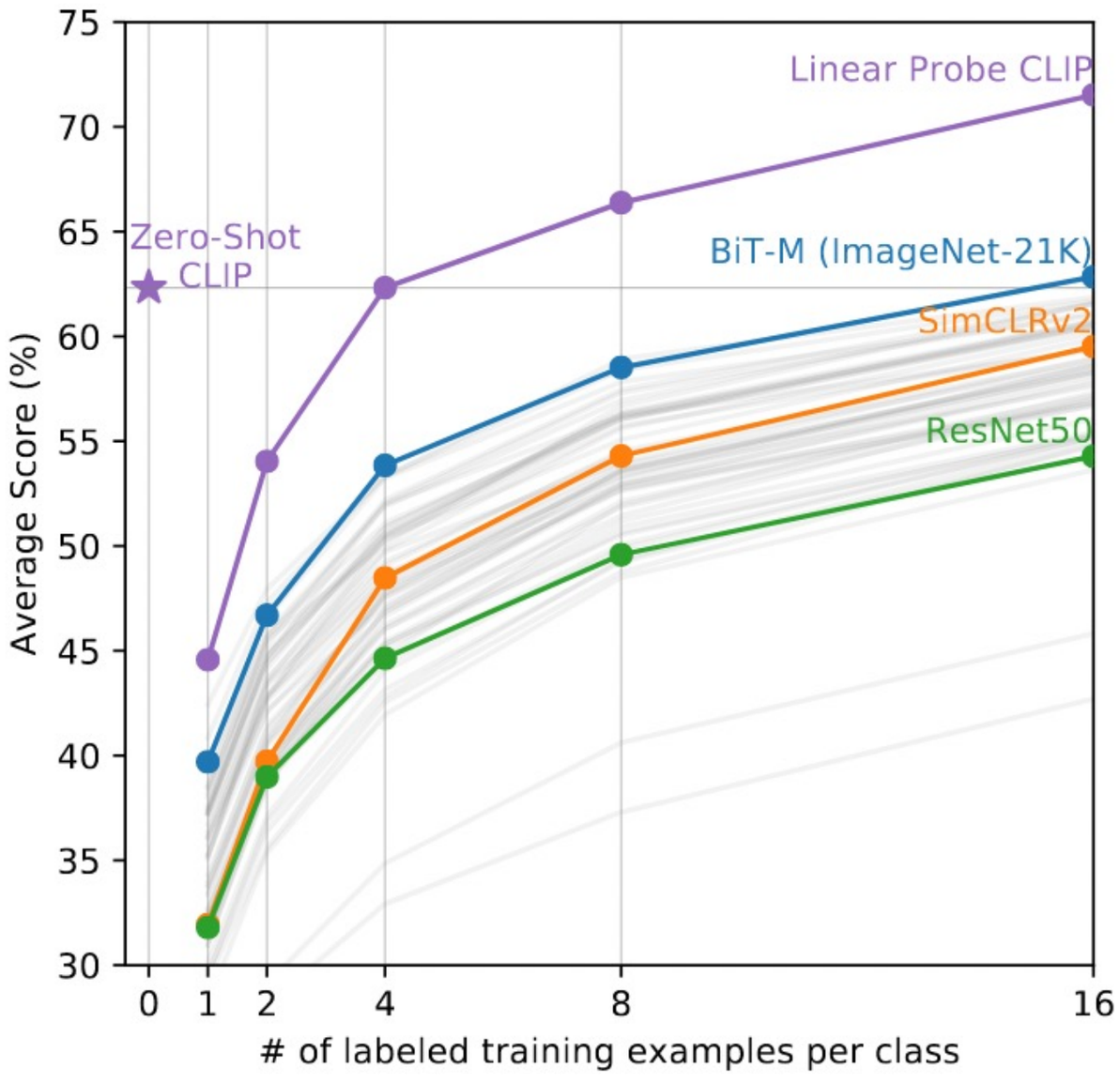


Figure 6. **Zero-shot CLIP outperforms few-shot linear probes.** Zero-shot CLIP matches the average performance of a 4-shot linear classifier trained on the same feature space and nearly matches the best results of a 16-shot linear classifier across publicly available models. For both BiT-M and SimCLRv2, the best performing model is highlighted. Light gray lines are other models in the eval suite. The 20 datasets with at least 16 examples per class were used in this analysis.

Human Performance

	Accuracy	Majority Vote on Full Dataset	Accuracy on Guesses	Majority Vote Accuracy on Guesses
Zero-shot human	53.7	57.0	69.7	63.9
Zero-shot CLIP	**93.5**	**93.5**	**93.5**	**93.5**
One-shot human	75.7	80.3	78.5	81.2
Two-shot human	75.7	85.0	79.2	86.1

Table 2. Comparison of human performance on Oxford IIT Pets. As in Parkhi et al. (2012), the metric is average per-class classification accuracy. Most of the gain in performance when going from the human zero shot case to the human one shot case is on images that participants were highly uncertain on. "Guesses" refers to restricting the dataset to where participants selected an answer other than "I don't know", the "majority vote" is taking the most frequent (exclusive of ties) answer per image.

Human jumps from zero to one-shot, indicating *humans know what they don't know* and are able to update their priors on the images they are most uncertain in based on a single example

Limitation

Thief, criminal, suspicious person classes

Animal, gorilla, chimpanzee, orangutan classes

Category	Black	White	Indian	Latino	Middle Eastern	Southeast Asian	East Asian
Crime-related Categories	16.4	24.9	24.4	10.8	19.7	4.4	1.3
Non-human Categories	14.4	5.5	7.6	3.7	2.0	1.9	0.0

Table 6. Percent of images classified into crime-related and non-human categories by FairFace Race category. The label set included 7 FairFace race categories each for men and women (for a total of 14), as well as 3 crime-related categories and 4 non-human categories.

Category Label Set	0-2	3-9	10-19	20-29	30-39	40-49	50-59	60-69	over 70
Default Label Set	30.3	35.0	29.5	16.3	13.9	18.5	19.1	16.2	10.4
Default Label Set + 'child' category	2.3	4.3	14.7	15.0	13.4	18.2	18.6	15.5	9.4

Table 7. Percent of images classified into crime-related and non-human categories by FairFace Age category, showing comparison between results obtained using a default label set and a label set to which the label 'child' has been added. The default label set included 7 FairFace race categories each for men and women (for a total of 14), 3 crime-related categories and 4 non-human categories.

Adding appropriate categories can reduce misclassification

Conclusion

- CLIP is designed to mitigate a number of major problems in standard deep learning approaches in computer vision
  - **Costly dataset** (e.g., ImageNet required 25K human annotators to annotate 14M images for 22K object categories)

Conclusion

- CLIP is designed to mitigate a number of major problems in standard deep learning approaches in computer vision
  - **Costly dataset**
  - **Narrow learning** (e.g., ImageNet model is good at predicting 1K categories but that's all it can do out of the box; if you need another tasks/classes, you need to build the dataset and fine-tune the model). In contrast, CLIP can be applied to a new task just by telling CLIP's encoder the names of the task's visual concepts.

Conclusion

- CLIP is designed to mitigate a number of major problems in standard deep learning approaches in computer vision
  - **Costly dataset**
  - **Narrow learning**
  - **Poor real-world performance:** some models rely on spurious correlations to do well on a task (kind of like “cheating”, only optimizing performance on the benchmark, so will often perform badly when deployed in the wild). In contrast, CLIP is not trained on a particular benchmark so can’t cheat. When it is trained on ImageNet, it can improve its performance further on the task by almost a whooping 10%.

Take aways

- CLIP is highly efficient, more efficient than other models trained to predict *exact* text given images
- CLIP is flexible and general, learn a wide range of visual concepts directly from natural language, can do zero-shot performance on many tasks, and performance is more representative to real world, in-the-wild performance
- Using task-agnostic pre-training on internet scale natural language, which is powering NLP, CLIP shows that such pre-training can be leveraged to improve performance of deep learning for other fields

Limitation

- CLIP struggles on abstract/systematic tasks/fine-grained classification although it performs well on recognizing common objects
- CLIP has poor generalization to images not covered in its pre-training data (e.g., MNIST)
- CLIP can be sensitive to wording/phrasing and sometimes require trial and error “prompt engineering”
- CLIP can perpetuate bias in its pre-training data

Cultural and Geographical Influences on Image Translatability of Words across Languages

Nikzad Khani, Isidora Tourni, Mohammad Sadegh Rasooli, Chris Callison-Burch, Derry Wijaya

NAACL 2021

https://aclanthology.org/2021.naacl-main.19/

INDONESIAN - DETECTED

ENGLISH

serangan fajar

14 / 5000

dawn attack

30

Google

serangan fajar

Warga Siap Terima Serangan Fajar | Detik ...

detiksumsel.com

SERANGAN FAJAR, UJIAN BAWASLU - SKH Me...

mediasulut.co

Antisipasi Serangan Fajar, Bawaslu Akan Hidupkan ...

nasional.okezone.com

Niat Serangan Fajar, Amplop Caleg Ma...

Berita Serangan Fajar Hari Ini - Kabar Terbaru Ter...

Siapa yang Berani Laporkan Ada Serangan Fajar Bak...

Google money politics

Money Controls Politics Now More Than ...
dailycaller.com

Money Politics' Cases Discovered Days ...
jakartaglobe.id

Glove and Boots - Money, Politics, and ...
facebook.com

money politics agreement Stock Photo ...
alamy.com

Big Money Out of Politics ...
medium.com

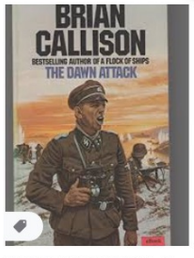
Michael Bennet: Congress must act on ...
dailycamera.com

Google dawn attack

Dawn Attack - TV Tropes
tvtropes.org

WFB: Dawn Attack - Bell of Los...
belloflostsouls.net

Frank Frazetta - Dawn Attack Print ...
frazettagirls.com · In stock



THE DAWN ATTACK - K...
amazon.com

The Attack at Dawn | 37.40 | The ...
art.thewalters.org

The Attack at Dawn - Wikipedia
en.wikipedia.org

attack at dawn | bl360wh@gmail.com | Fli...
flickr.com

Google शादी

Up Marriage Guidelines Guests Limit News: Up Go…
amarujala.com

Coronavirus Wedding Guidelines in UP: शादी स…
jagran.com

कर्फ्यू में कैसे करें दिल्ली के अंदर शादी, ये है तरीका - India TV Hin…
indiatv.in

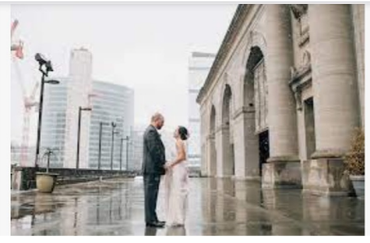
Uttar Pradesh Bride Ask To Groom Table Of Tw…
amarujala.com

इन 4 महीनों में नहीं है शादी का मुहूर्त, मई में सिर्फ 3 दिन बजेगी श…
aajtak.in

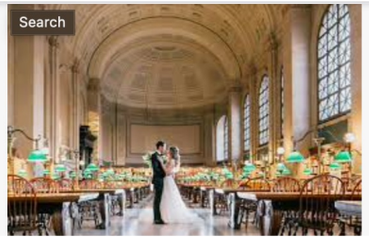
शादी के बाद ही सामने आती हैं ये 5 बातें - 5 things that happ…
aajtak.in

Google wedding

Here comes the bride: Wedding venues ...
masslive.com



Real Wedding: A Picture-Perfect ...
bostonmagazine.com



10 Historic Boston Wedding Venues ...
thecateredaffair.com

Emily + Casey | State Room Boston ...
lovelyvalentine.com

11 Restaurant Wedding Venues in Boston ...
bostonmagazine.com

2020 Weddings - 2020 Wedding Trends
harpersbazaar.com

MMID DATASET

98 languages

10,000 words per language

100 images per word

Languages Studied

- 19 language pairs were chosen such that every source language had two or more target languages with some shared characteristic in geography or culture.

Source	AZ	AZ	KO	KO	ZH	ZH	JA	JA	AR	AR	AR	UR	UR	ES	ES	FI	FI	AF	AF
Target	TR	RU	ZH	JA	JA	KO	ZH	KO	UR	FA	HE	AR	HI	FR	PT	HU	NO	NL	SW

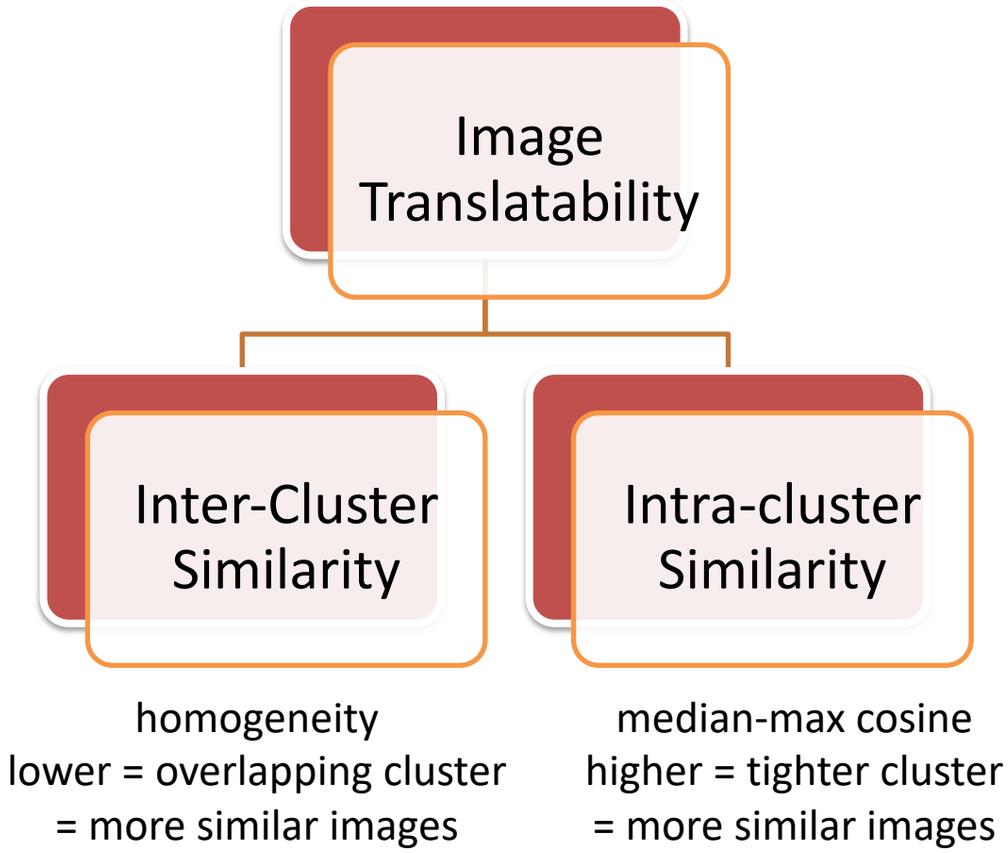
Language Key Codes: AZ – Azerbaijani, TR – Turkish, RU – Russian, KO – Korean, JA – Japanese, ZH – Chinese, AR – Arabic, UR – Urdu, FA – Farsi, HE – Hebrew, HI – Hindi, ES – Spanish, PT – Portuguese, HU – Hungarian, FI – Finnish, NO – Norwegian, AF – Afrikaans, NL – Dutch, SW – Swahili

Image embeddings

- For a given language pair, source words were taken from MMID, then translated to target with Google Translate
- We found both the source and target word's images in MMID
- Images passed through Res-Net 50, pre-trained on ImageNet
- Each image embedding is a 2048-length vector
- We call all 100 image embeddings for a word an image embedding cluster
- We call a source word's image embedding cluster and its translation's image embedding cluster a word pair

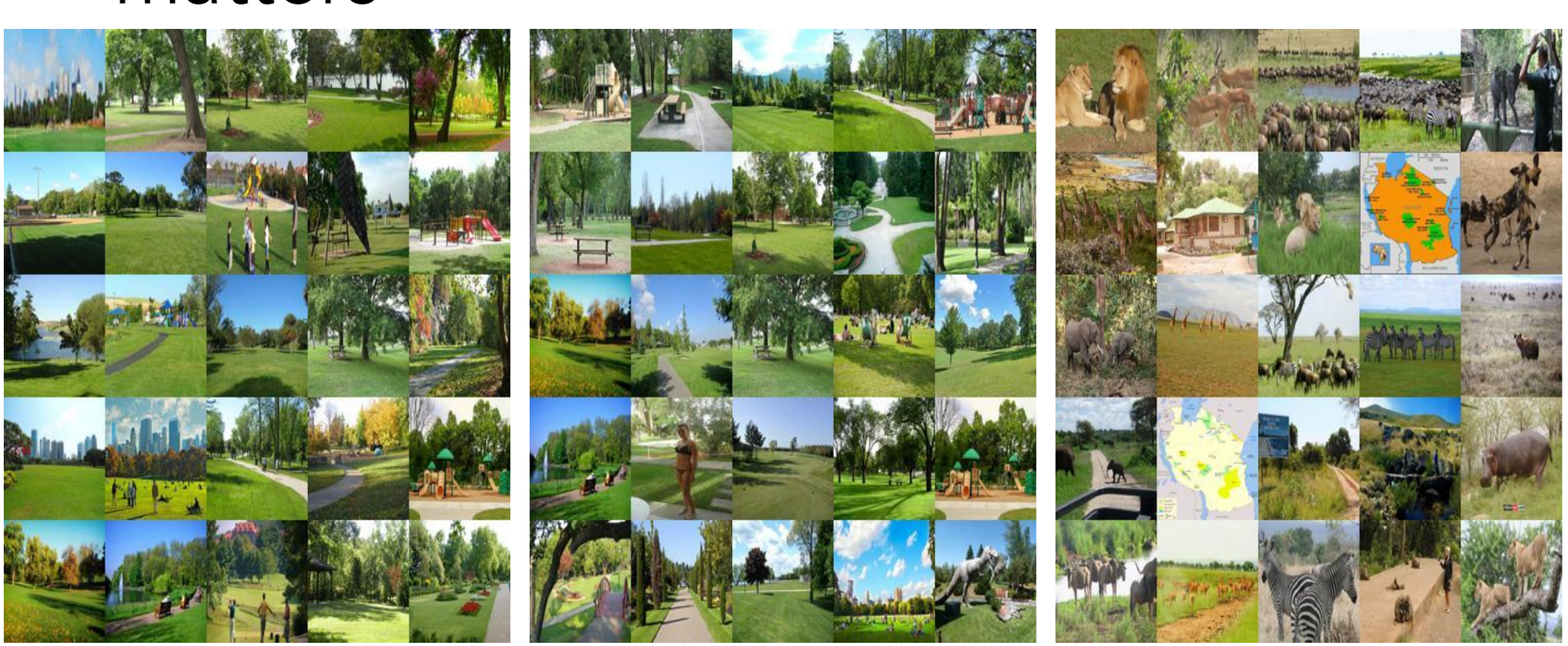
Image translatability as a metric

- Intra-Cluster Similarity is a measure of how tight a set of images are. In other words, how diverse are the images for one search result.
- Inter-Cluster Similarity is a measure of how different the two sets of images are. In other words, the similarity of the source word's search results and its translation's search results.
- High Image Translatability means high inter and intra-cluster similarity



Language and Vision in Translation

- Typological similarity between languages matters



Images of the word “park” in Afrikaans, Dutch, and Swahili

Massively Multilingual Image Dataset (MMID)

100
Languages

10,000
Words per language

250,000
English word translations

100
Images per word

35,000,000
Total images

20TB
of data
Hosted by Amazon Public Datasets multilingual-images.org

Translatability of Words via Images

- We measure similarities of image representations of words that are translations of each other
- Language pairs with shared culture have improved image translatability, regardless of geographic proximity

Language Pair		Ratio	
		All	Concrete
az	**tr**	**0.31**	**0.37**
az	ru	0.17	0.22
ar	ur	0.39	0.44
ar	fa	0.50	0.55
ar	**he**	**0.69**	**0.73**
ur	**ar**	**0.39**	**0.45**
ur	hi	0.12	0.15
es	fr	0.45	0.58
es	pt	0.40	0.53
fi	**hu**	**0.29**	**0.40**
fi	no	0.17	0.26
af	**nl**	**0.39**	**0.50**
af	sw	0.25	0.31

Conclusion

Cultural similarities > geographical proximity between language pairs

Concrete words provide better image translatability

Image-aided machine translation can provide better context