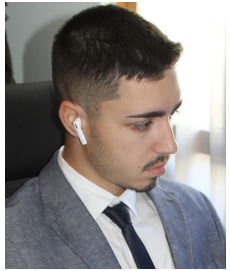
Sapienza Training Camp 2021

Building an Image Search Engine

2 - 4 September, 2021

Instructor team

Dr. Mykhaylo Andriluka
Google Research, Zurich



Alessio Sampieri
Sapienza University
of Rome

Laura Laurenti
Sapienza University
of Rome

Prof. Dr. Fabio Galasso
Sapienza University
of Rome

Organization

- Website: sapienza-training-camp2021.github.io
- Getting started doc (link)
- Ask questions on Slack

- Course format:
  - Competition on Kaggle
  - Lectures: 15-20 minute blocks with quiz in the end

Image search engine

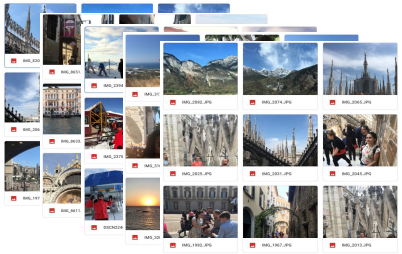


Photo collection
(e.g. pictures from the last vocation or a database of web images)

Query: “Dog in glasses sitting in a basket on a bike”

Image search engine

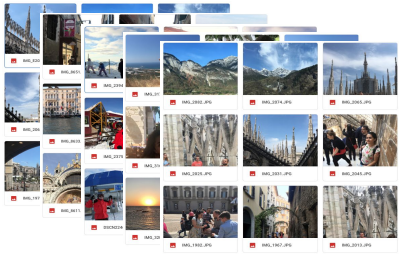


Photo collection
(e.g. pictures from the last vocation or a database of web images)

Search engine

Query: “Dog in glasses sitting in a basket on a bike”

Result: images in the collection ordered by relevance with respect to the query

Image search engine

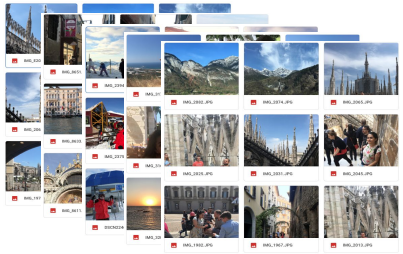
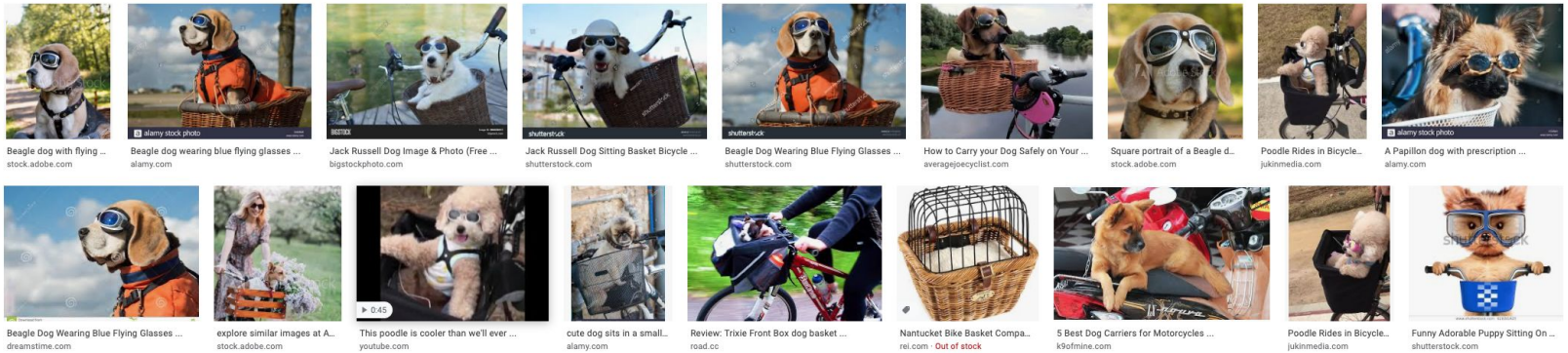


Photo collection
(e.g. pictures from the last vocation or a database of web images)

Search engine

Query: “Dog in glasses sitting in a basket on a bike”



Result: images in the collection ordered by relevance with respect to the query

Image search engine

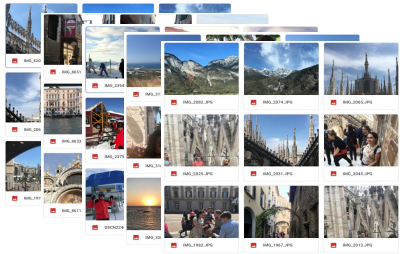
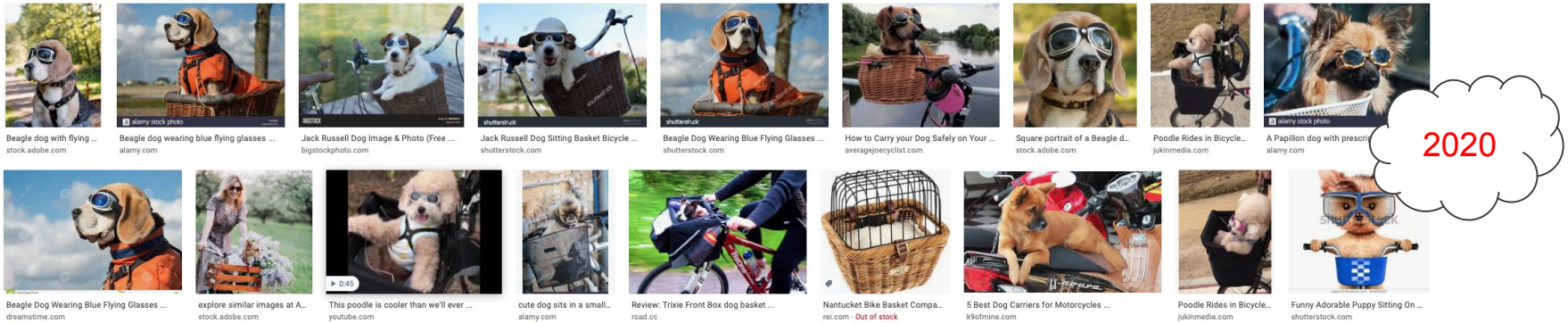


Photo collection
(e.g. pictures from the last vocation or a database of web images)

Search engine

Query: “Dog in glasses sitting in a basket on a bike”



Result: images in the collection ordered by relevance with respect to the query

Image search engine

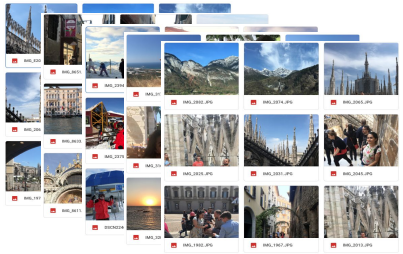
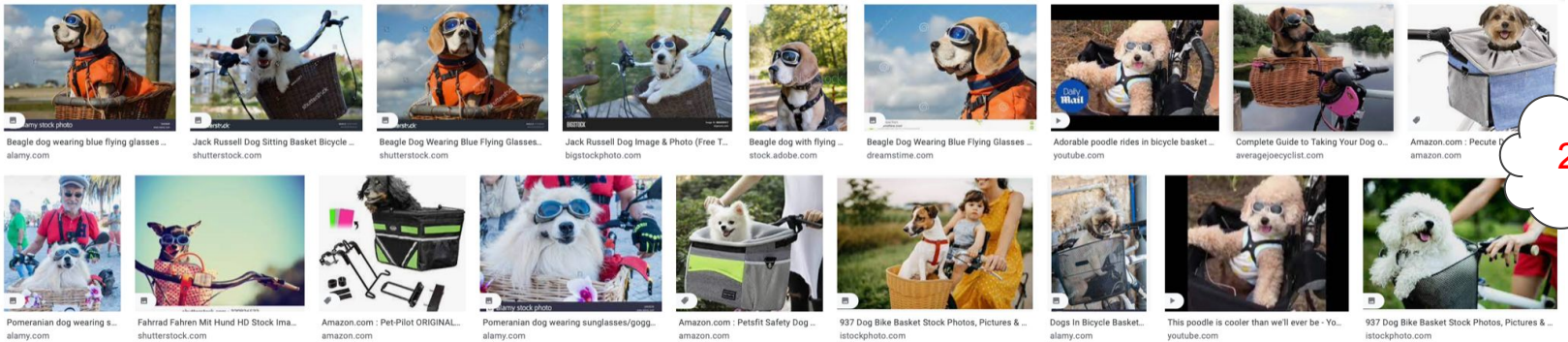


Photo collection
(e.g. pictures from the last vocation or a database of web images)

Search engine

Query: “Dog in glasses sitting in a basket on a bike”



Result: images in the collection ordered by relevance with respect to the query

Disclaimer: we will skip a lot of details

- Skip a lots of technical aspects related to implementation of the real search engine
  - e.g. how to build a system that can store and index terabytes of data
- Focus on image content only
  - ignore the location where the photograph has been taken
  - ignore text of the webpage that contains the image
- Consider each photograph individually
  - we won’t build models for places or specific people

Focus on “learning by doing”

- You will develop your own version of the “search engine” and participate in the in-class Kaggle competition (more about it later today)
- Lectures will closely follow the code you will use in your implementation
  - experiment with the code to learn what works best
- Lecture followed by a quiz to recall what you learned
  - quizzes are for you to test yourself (can take quiz multiple times)
  - best strategy: take a quiz after the lecture and then again 2-3 days later (test your memory)

Roadmap

Image from the photo collection

I_k

Q

Query: “dog in glasses sitting in a basket on a bike”

Roadmap

Image from the photo collection

I_k

Neural Network (CNN)

Image encoding

φ_k

Q

Query: “dog in glasses sitting in a basket on a bike”

Roadmap

Image from the photo collection

I_k

Neural Network (CNN)

Image encoding

φ_k

Recurrent neural network (RNN)

Image description

T_k

= “dog next to person on the bike near street crossing”

Q

Query: “dog in glasses sitting in a basket on a bike”

Roadmap

Image from the photo collection

I_k

Neural Network (CNN)

Image encoding

φ_k

Recurrent neural network (RNN)

Image description

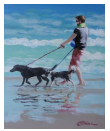
T_k

= “dog next to person on the bike near street crossing”

Q

Query: “dog in glasses sitting in a basket on a bike”

Define similarity function. Order images according to similarity to the query.



$$\text{sim}(Q, T_1) > \text{sim}(Q, T_2)$$

Day 1

Image from the photo collection

I_k

Neural Network (CNN)

Image encoding

φ_k

Recurrent neural network (RNN)

Image description

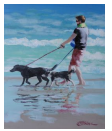
T_k

= “dog next to person on the bike near street crossing”

Q

Query: “dog in glasses sitting in a basket on a bike”

Define similarity function. Order images according to similarity to the query.



$$\text{sim}(Q, T_1) > \text{sim}(Q, T_2)$$

Day 2

Image from the photo collection

I_k

Neural Network (CNN)

Image encoding

φ_k

Recurrent neural network (RNN)

Image description

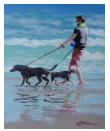
T_k

= “dog next to person on the bike near street crossing”

Q

Query: “dog in glasses sitting in a basket on a bike”

Define similarity function. Order images according to similarity to the query.



$$\text{sim}(Q, T_1) > \text{sim}(Q, T_2)$$

Day 2

Image from the photo collection

I_k

Neural Network (CNN)

Image encoding

φ_k

Recurrent neural network (RNN)

Image description

T_k

= “dog next to person on the bike near street crossing”

Q

Query: “dog in glasses sitting in a basket on a bike”

Define similarity function. Order images according to similarity to the query.

$$\text{sim}(Q, T_1) > \text{sim}(Q, T_2)$$

Day 3

Image from the photo collection

I_k

Neural Network (CNN)

Image encoding

φ_k

Recurrent neural network (RNN)

Image description

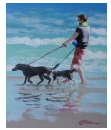
T_k

= “dog next to person on the bike near street crossing”

Q

Query: “dog in glasses sitting in a basket on a bike”

Define similarity function. Order images according to similarity to the query.



$$\text{sim}(Q, T_1) > \text{sim}(Q, T_2)$$

Course schedule

Day 1 (Thursday, 2nd of September)

Module	Time	Quiz	Additional material	Slides
Introduction to the training camp	10:00 - 10:30	-	Machine learning with Tensorflow2, Books: Deep learning, Deep learning with Python	
Neural networks recap	10:30 - 11:30	-	-	-
Convolutional neural networks	-	-	-	-
Introduction to the Kaggle competition	13:30 - 14:00	-	-	-
"TensorFlow, Keras and deep learning, without a PhD" - Q&A session	14:00 - 15:00	-	-	-

Day 2 (Friday, 3rd of September)

Module	Time	Quiz	Additional material	Slides
CNN Architectures: VGG, ResNet, Inception, MobileNet	13:30 - 15:00	-	Deep Residual Learning for Image Recognition, Going Deeper with Convolutions	-
Neural Networks for Natural Language Processing (NLP)	-	-	Visualizing and Understanding Recurrent Networks, A Simple but Tough-to-Beat Baseline for Sentence Embeddings, How image search works at Dropbox	-

Day 3 (Saturday, 4th of September)

Module	Time	Quiz	Additional material	Slides
Image captioning. Attention models in computer vision and NLP.	10:00 - 11:30	-	Show, Attend and Tell: Neural Image Caption Generation with Visual Attention, Image captioning colab	-
End of the Kaggle competition.	16:00	-	-	-
Announcement of the Kaggle competition results. Short presentations by the competition winners.	17:00 - 17:30	-	-	-

for up-to-date schedule check the course webpage
sapienza-training-camp2021.github.io

Kaggle competition

- Competition is hosted on Kaggle (www.kaggle.com)
- You can participate in teams of at most 3 people
  - We strongly encourage you to work in a team
  - Let us know if you can not find a team yourself

- Saturday 4pm: end of the competition, winner announcement, short presentations by the winning teams

Kaggle competition

- Competition is based on the COCO Dataset: https://cocodataset.org/#explore

- The COCO dataset provides a set of images and their textual descriptions:

a cat laying on the keyboard of a computer.
a cat laying on top of a laptop computer keyboard.
a cat that is laying on top of a laptop.
a cat sitting across the keyboard of a computer
a cat sits on top of a laptop computer.

a large three layered cake with yellow filling sliced on a white plate
a cake on a plate, on the ground, with four slices cut.
a close up of food on a plate being cut into slices
someone has begun to cut the cake into slices.
a cake with white icing being sliced with a knife.

Kaggle competition

- In the competition you will be given a set of images and a corresponding set of textual queries
- Your task:
  - compare query with each of the images
  - generate list of images sorted by similarity to the query = implement image search!
- You need to submit a valid competition entry to pass
- Don’t panic! We will provide you with a starting code package :)
- More details today at 1:30pm

Looking forward to the next three days!

Good luck for the competition!

Roadmap

Image from the photo collection

I_k

Neural Network (CNN)

Image encoding

φ_k

Recurrent neural network (RNN)

Image description

T_k

= “dog next to person on the bike near street crossing”

Q

Query: “person walking with a dog on the beach”

Define similarity function. Order images according to similarity to the query.

$$\text{sim}(Q, T_1) > \text{sim}(Q, T_2)$$

Roadmap

Image from the photo collection

I_k

Neural Network (CNN)

Image encoding

φ_k

Recurrent neural network (RNN)

T_k

Image description

= “dog next to person on the bike near street crossing”

Q

Query: “person walking with a dog on the beach”

Define similarity function. Order images according to similarity to the query.

$$\text{sim}(Q, T_1) > \text{sim}(Q, T_2)$$

Roadmap

Image from the photo collection

I_k

→

Colab:

https://github.com/SapienzaTrainingCamp/GoogleTrainingCamp/blob/main/Notebook/1-ImageCaptioningBase.ipynb

→

Image description

T_k

= “dog next to person on the bike near street crossing”

Q

Query: “person walking with a dog on the beach”

Define similarity function. Order images according to similarity to the query.

$$\text{sim}(Q, T_1) > \text{sim}(Q, T_2)$$

Roadmap

Image from the photo collection

I_k

Neural Network (CNN)

Image encoding

φ_k

Recurrent neural network (RNN)

Image description

T_k

= “dog next to person on the bike near street crossing”

Q

Query: “person walking with a dog on the beach”

Define similarity function. Order images according to similarity to the query.

$$\text{sim}(Q, T_1) > \text{sim}(Q, T_2)$$

How to compare images and text?

Image from the photo collection

I_k

Q

Query: “person walking with a dog on the beach”

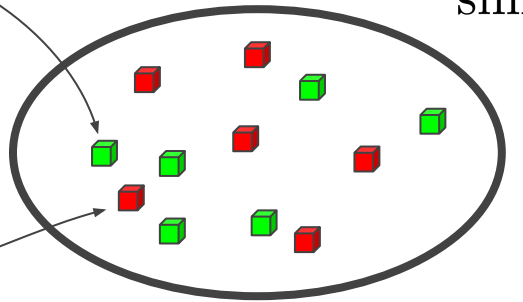
How to compare images and text?

Image from the photo collection

I_k

We need to introduce a **common representation** for images and text to measure their similarity

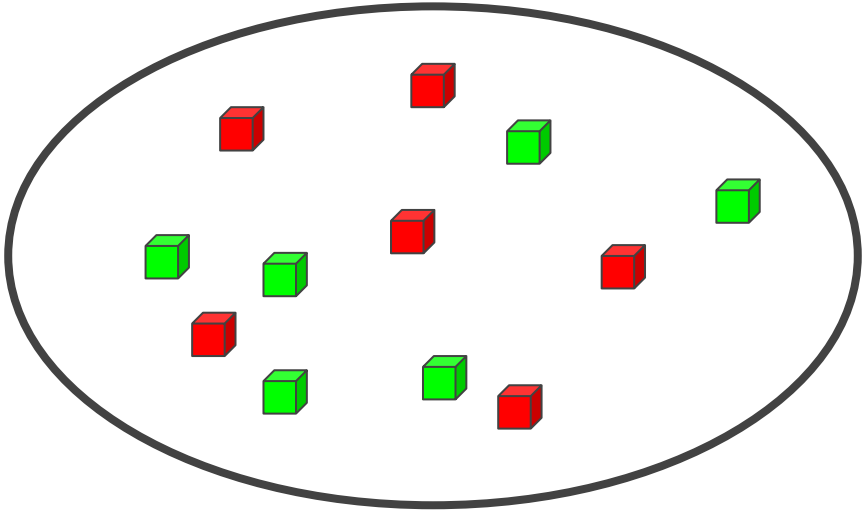
$\mathrm{sim}($  ,  $)$



Q

Query: “person walking with a dog on the beach”

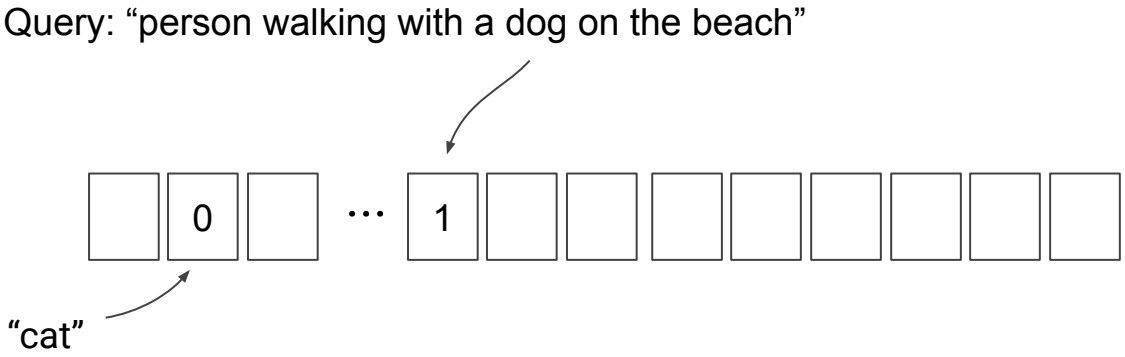
Embedding space



Path to success: experiment with different text and image representations

Text embeddings

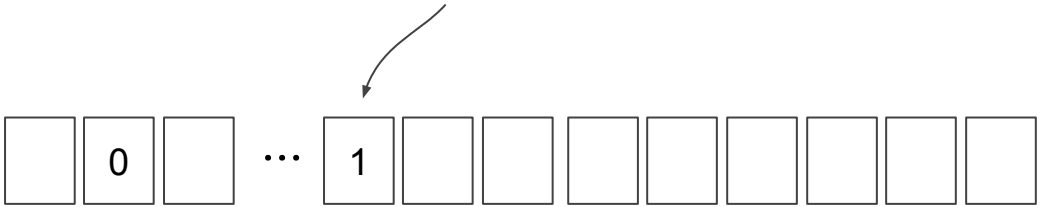
- One-hot vectors = very simple embedding



Text embeddings

- One-hot vectors = very simple embedding

Query: “person walking with a dog on the beach”



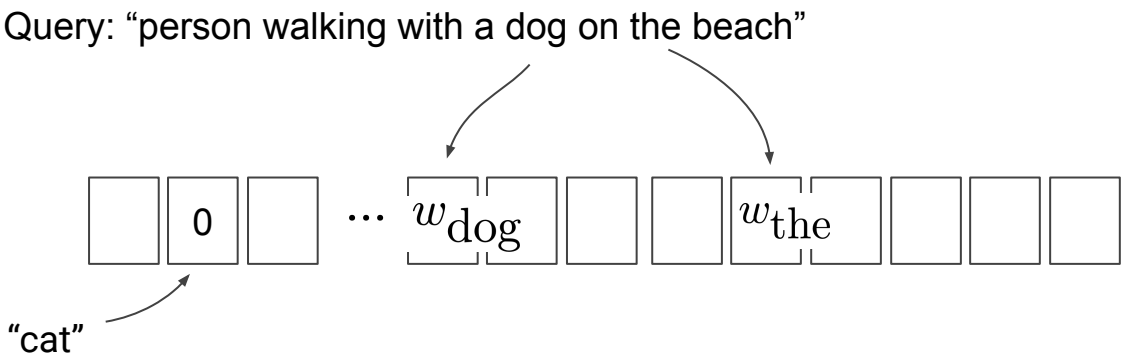
one dimension for each word in the vocabulary

$$\text{sim}(\mathbf{e}_1, \mathbf{e}_2) = \frac{\mathbf{e}_1^T \mathbf{e}_2}{\|\mathbf{e}_1\| \|\mathbf{e}_2\|}$$

cosine similarity (link), works for any embedding

TF-IDF weighting

- Problem: some words are more informative than others



w_{dog} - TF-IDF weight (https://en.wikipedia.org/wiki/Tf–idf)

Learned word embeddings

- Problem:
  - some words have different meaning on the context
  - different words have similar meaning
  - how to capture the meaning of a word in a vector?
- Solution: learned word embeddings
  - http://web.stanford.edu/class/cs224n/slides/cs224n-2021-lecture01-wordvecs1.pdf
- Pre-trained word embedding models
  - GloVe
  - BERT
  - See "Student instructions" for references on how to download and use these word embedding models
- Sentence embeddings
  - average words in a sentence
  - more advanced models available (link1, link2)

Image representation

Image from the photo collection

I_k

Neural Network (CNN)

Image encoding

φ_k

Recurrent neural network (RNN)

Image description

T_k

= “dog next to person on the bike near street crossing”

Image representation

Image from the photo collection

I_k

Neural Network (CNN)

Image encoding

φ_k

Recurrent neural network (RNN)

Image description

T_k

= “dog next to person on the bike near street crossing”

Representation 1

Image representation

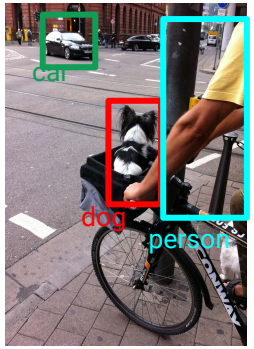
Image from the photo collection

I_k

Neural Network (CNN)

Image encoding

φ_k



Representation 2

Publicly available object detectors: “Detectron2”, “Object Detection API”, ...

Image representation

Image from the photo collection

I_k

Neural Network (CNN)

Image encoding

φ_k

Image classifier:
[dog: 0.9, pavement: 0.8, grass:0.01, ….]

Representation 3

See “How image search works at Dropbox”

Image representation

Image from the photo collection

I_k

Neural Network (CNN)

Image encoding

φ_k

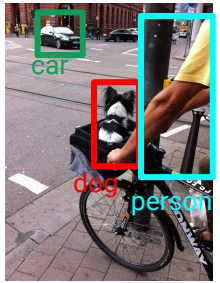
Recurrent neural network (RNN)

Image description

T_k

= “dog next to person on the bike near street crossing”

Representation 1



Representation 2

Image classifier:
[dog: 0.9,
pavement: 0.8,
grass:0.01, ….]

Representation 3

Take a quiz!