Zastosowanie sieci CNN uczonych na zaszumionych danych do klasyfikacji zdjęć

Architecture – VGG16

Original network had 3 FC layers at the end: 2 of those had 4096 neurons and the third last layer had as much layers as the number of categories.

My net has 4 FC layers at the end: 3 of those have 512 neurons and the last has 10 as the number of cities in the dataset.



Architecture

Modified:

- Fully connected layers: 512, 512, 512, 10
- Batch size: 4

Original:

- Fully connected layers: 4096, 4096, 10
- Batch size: 64



Architecture change



Original VGG16 was extended by adding regularization to enable longer training without overfitting. Regularization was introduced in the form of dropout after the fully connected layers with 4096 neurons. The magnitude of dropout was set to 0.5 which means at each iteration 50% of neurons were randomly dropped.

Training – procedure used previously



There are two main blocks of layers within CGG16 network: **feature extraction layers** which are responsible for extracting feature meaningful for the problem at hand and **problem solving layers** which are aimed at solving the problem.

Till now the training procedure used weights in **feature extraction layers** as the were (trained on ImageNet dataset containing 1k common objects) without changing them and only modified the **problem solving layers**. Training all weights was unsuccessful till now.

Training – procedure change



The current approach first trains the problem solving layers up to its maximum potential and once the validation error plateaus trains all layers (both feature extraction layers and problem solving layers) at a decreased learning rate. Providing a good solution as a starting point of the fine tuning process proves successful and the network is able to achieve a much higher results.

This was not correct: I froze all layers except for the last 3 ones (according to the original net design), but since I added regularization layers the last 3 layers were: SoftMax(10), Dropout(0.5) and Fully Connected(4096). This is not the entire problem solving group of layers.

Training – Comparison

Old approach:



Feature extraction layers + 2 layers: FROZEN Problem Solving Layers: TRAINABLE Learning rate: 0.0001

New approach:



Feature extraction layers + 2 layers: FROZEN Problem Solving Layers: TRAINABLE Learning rate: 0.0001



ADJUSTED slide.



Feature extraction layers + 2 layers: TRAINABLE Problem Solving Layers: TRAINABLE Learning rate: 0.00001

Different ways of fine tuning comparison

First approach (originating in code error)





Feature extraction layers + 2 layers: FROZEN Problem Solving Layers: TRAINABLE Learning rate: 0.0001

Second approach (aligned with theory)



Feature extraction layers: FROZEN Problem Solving Layers: TRAINABLE Learning rate: 0.0001





Feature extraction layers + 2 layers: TRAINABLE Problem Solving Layers: TRAINABLE Learning rate: 0.00001

3x3 conv, 64	3x3 conv, 64	pool2	3x3 conv. 128	+	3x3 conv, 128	pool/2	+	3x3 conv, 256	+	3x3 conv, 256	-	3x3 conv, 256	2/Jood	3x3 conv. 512	T	3x3 conv, 512	-	3x3 conv, 512	pool2	3x3 conv, 512	+	3x3 conv. 512	-	3x3 conv, 512	pool/2	fc 4096	•	Dropout 0.5	-	fc 4096	-	Dropout 0.5	
ize:224		J	ize112			J	C	ize:56		_	1 ()	ize:28	9)	izer14)	dzer7)		

Feature extraction layers: TRAINABLE Problem Solving Layers: TRAINABLE Learning rate: 0.00001

Different ways of fine tuning comparison – results comparison



Second approach (aligned with theory)



Experiment 1:	Experiment 2:	Experiment 3:	Experiment 1:	Experiment 2:
Random:	Random:	Random:	Random:	Random:
Accuracy	Accuracy	Accuracy	Accuracy	Accuracy
mean 0.609867	mean 0.610033	mean 0.609433	mean 0.516567	mean 0.514300
std 0.007698	std 0.005390	std 0.005622	std 0.006959	std 0.007438
Test:	Test:	Test:	Test:	Test:
0.87	0.87	0.87	0.77	0.77

The results show that the error in the code lead to a much better result.

Food dataset

Food dataset - creation

List of top 10 foods:

- <u>https://visual.ly/community/infographic/food/top-10-americas-favorite-foods</u>
- https://food.ndtv.com/food-drinks/10-american-foods-777850
- <u>http://islandgrownschools.weebly.com/uploads/1/0/7/8/10785576/top_ten_foods_consumed_in_america.pdf</u>

Selected list of top 10 food is a mixture of the above sources to manage various restriction of training (popularity of hashtag on Instagram) and test (existence in food-101 data set) data availability. This list focuses more on America because the bias of Instagram.

- 1. Apple pie
- 2. Burger
- 3. Donuts
- 4. French Fries
- 5. Hot Dog
- 6. Macaroni and cheese
- 7. Pancake
- 8. Pizza
- 9. Spaghetti
- 10. Steak

Food dataset - creation

Data set split:

- Instagram Data 800k images downloaded from Instagram containing one of the hashtags from the list of top 10 food. This data is divided into:
 - Training data 770k images from 10 categories (equal number of images from each category)
 - Random testing data 30k images from 10 categories (equal number of images from each category)
- Independent test data 3k images from 10 categories (equal number of images from each category). This data comes from Kaggle and it was verified to contain one of the top 10 food.

Experiment hypothesis:

Once trained on noisy web data (not sure if class truly appears) we assume that the net will be able to categorize previously not seen NOT NOISY data with high accuracy. We want to validate the hypothesis by comparing results achieved for randomly selected datasets from Instagram that did not take part in the training procedure with independent test data where we know that the class appears.

Food dataset – is it characteristic?



Food dataset – noisy data reminder

Webly data

There are various reasons why data associated with a particular hashtag might be incorrect:

- Label does not correspond to reality
- There are more than one class on the image
- The image is of low quality

Below there are examples of the following categories: apple pie, burger and pancake.





Food dataset – VGG16

Problem solving training:

Fine tuning:



Problem	solving	training:

Random:							
	Accuracy						
mean	0.448800						
std	0.007996						
Test:							
0.688							

Experiment 1:

Experiment 2	<u>2:</u>	Experiment 3:				
Random: Accura mean 0.4570 std 0.0060 Test: 0.7	acy 300 509	Rando mean std Test: 0.69	m: Accuracy 0.45490 0.00573			

Accuracy 0.454900

0.005737

	Random:	Random:	Random:
Fine tuning:	Accuracy	Accuracy	Accuracy
	mean 0.609867	mean 0.610033	mean 0.609433
	std 0.007698	std 0.005390	std 0.005622
	Test:	Test:	Test:
	0.87	0.87	0.87

Results – in-depth analysis

Fine tuning:

Test:		
0.87		
Test by vategory:		
Acci	uracy	
Categ	-	
burger	0.64	
applepie	0.75	
donuts	0.84	
hotdog	0.86	
steak	0.87	
macandcheese	0.89	
pancake	0.90	
pizza	0.94	
spaghetti	0.98	
frenchfries	0.98	
Confusion matrix,	without	normalization

Experiment 1:

Experiment 2:

Test: Test: 0.87 0.87 Test by vategory: Test by vategory: Accuracy Categ Categ burger burger 0.65 applepie applepie 0.77 hotdog hotdog 0.83 steak steak 0.84 donuts donuts 0.89 macandcheese 0.90 pancake pancake 0.93 pizza pizza 0.94 frenchfries frenchfries 0.98 spaghetti spaghetti 0.98 Confusion matrix, without normalization

0.74 0.84 0.85 0.86 macandcheese 0.90 0.93 0.95 0.98

0.65

Accuracy

Experiment 3:

0.99 Confusion matrix, without normalization













Food dataset – ResNet

Fine tuning:



ResNet was able to achieve the same level of accuracy in a shorter time span. It did not require a 2 stage training process (problem solving layers training, fine tuning). This result was achieved training all neurons since the beginning at a learning rate of 0.00001 (the same as for fine tuning in VGG16).

Results – in-depth analysis

Random:

Experiment 1:

Experiment 2:

Experiment 3:

Accuracy

0.66

0.76

0.85

0.85

Fine tuning:

Accuracy mean 0.604800 std 0.007096 Test: 0.88 Test by vategory: Accuracy Categ 0.69 burger applepie 0.74 hotdog 0.86 0.88 steak donuts 0.88 macandcheese 0.89 0.92 pancake pizza 0.96 frenchfries 0.97 spaghetti 0.98

Rando	m:			
	Acci	uracy		
mean	0.6	05133		
std	0.0	07962		
Test:				
0.86				
Test	by va	ategory:		
		Acci	uracy	
Categ				
burge	r		0.63	
apple	pie		0.72	
steak	:		0.81	
hotdo	g		0.83	
donut	s		0.86	
macan	dche	ese	0.90	
panca	ke		0.93	
pizza	1		0.95	
frenc	hfri	es	0.98	
spagh	etti		0.98	
Confu	sion	matrix,	without	norm

applepie

burger

donuts

hotdog - 1

pancake - 31

spaghetti

steak

pizza 5 2

frenchfries

macandcheese

label

Predicted

4

6 259 0

7 7

17 3 3 0

3

12 11 4

-5 0 1

Confusion matrix, without normalization

0

8 2 4

0 4

3

3 73 6 294 20

0 1 4

691155000

6 2

4

4 1

4

6 0

True label

7 1 3 0 0

2 1 3 0 0 0

5001111

6

- 4

3 0 13

5 3 18

284 0 3

4 11

278 2 0 6

0.88 donuts macandcheese 0.89 pancake 0.91 pizza 0.95 spaghetti 0.97 frenchfries 0.98

Accuracy

mean 0.605867 0.006317

Test by vategory:

Random:

std

Test:

0.87

Categ

burger

hotdog

steak

- 250

- 200

- 150

100

- 50

0

applepie

Confusion matrix, without normalization





Instacities dataset

Food dataset - creation

Data set split:

- Instagram Data 800k images downloaded from Instagram containing one of the hashtags from the list of 10 cities. This data is divided into:
 - Training data 770k images from 10 categories (equal number of images from each category)
 - Random testing data 30k images from 10 categories (equal number of images from each category)
- Independent test data this data comes from official Instagram accounts of the cities in training set. The list of
 accounts is presented below. Each account has a various number of images. We have constructed 2 test sets from
 those images one of random 300 images per category and the other with images that we believe are
 characteristic for the city (like "Big Ben" for London).
 - @chicago
 - @cityofmelbourne
 - @london
 - @losangeles_city
 - @nycgov
 - @onlyinsf
 - @seetorontonow
 - @sydney
 - @visit_singapore

Food dataset – is it characteristic?





Instacities dataset – VGG16

The process for training VGG16 net for Instacities dataset was a bit more complex. Eventually I used a setup with 5 stages but it could probably be reduced to 4 or less.

Stage	Trainable layers	Learning Rate
Stage 1	Last 3	1e-4
Stage 2	Last 3	1e-5
Stage 3	Last 3	1e-6
Stage 4	All	1e-5
Stage 5	All	1e-6

The majority of knowledge extraction and the biggest improvement can be seen in stages 1 and 4 which initiate learning some of the layers.





Results – in-depth analysis – all test cases

Random:

Experiment 1:

Fine tuning:

Nanuolii.	
Accuracy	
mean 0.303167	
std 0.007492	
Test:	
0.45	
Test by vategory	·:
Ac	curacv
Categ	,
toronto	0.26
losangeles	0.28
newvork	0.20
melhounne	0.00
meibourne	0.39
sanfrancisco	0.42
london	0.46
singapore	0.47
sydney	0.50
chicago	0 62



Experiment 2:

Random:

	Accurac	У
mean	0.30050	0
std	0.00671	7
Test:		
0.45		
Test	by vateg	ory:
		Accuracy
Categ		
toron	to	0.26
losan	geles	0.28
melbo	urne	0.34
newyo	rk	0.35
sanfr	ancisco	0.40
londo	n	0.47
singa	pore	0.51
sydne	y	0.52
chica	go	0.60



Experiment 3:

Random:	
Accuracy	
mean 0.300767	
std 0.008439	
Test:	
0.46	
Test by vategory	:
Ac	curacy
Categ	
toronto	0.28
losangeles	0.31
melbourne	0.32
newyork	0.35
sanfrancisco	0.42
london	0.46
singapore	0.53
sydney	0.53
chicago	0.65



Results – in-depth analysis – 300 from test

	<u>Experime</u>	nt 1:	Experi	<u>ment 2:</u>	Experi	<u>ment 3:</u>	
LO x Randomly Test Random: 0.42 (+/-0.0073)		Test Random: 0.42 (+/-0.0079)		Test Random: 0.43 (+/-0.009	2)		
selected 300	Test Random by chicago	category MEAN: 0.62	Test Random by chicago	category MEAN: 0.61	Test Random by chicago	category MEAN: 0.65	
from test	<pre>london losangeles melbourne</pre>	0.46 0.28 0.39	london losangeles melbourne	0.47 0.28 0.34	london losangeles	0.46 0.30	
<u>images:</u>	newyork sanfrancisco singapore	0.38 0.41 0.47	newyork sanfrancisco	0.35 0.41	newyork sanfrancisco	0.35 0.43	
	sydney toronto dtype: tloat64	0.50 0.25	singapore sydney toronto	0.52 0.26	sydney toronto	0.55 0.53 0.28	
	Test Random by chicago	category STD: 0.0276	Test Random by chicago	category STD:	Test Random by chicago	category STD:	
	losangeles melbourne	0.0219 0.0237	london losangeles melbourne	0.0280 0.0228 0.0219	london losangeles melbourne	0.0292 0.0246 0.0241	
	newyork sanfrancisco singapore	0.0144 0.0269 0.0242	newyork sanfrancisco singanore	0.0146 0.0273 0.0238	newyork sanfrancisco singanore	0.0151 0.0253 0.0253	
	sydney toronto	0.0266 0.0238	sydney toronto	0.0285 0.0217	sydney toronto	0.0280	
Selected 300	Test Selected: 0.71		dtype: float64 Test Selected: 0.69		dtype: float64 Test Selected: 0.7		
from test	m test Selected by vategory: Accuracy		Test Selected by vategory: Accuracy		Test Selected by vategory: Accuracy		
<u>images:</u>	Categ chicago london sydney losangeles melbourne sanfrancisco toronto newyork	0.58 0.59 0.69 0.72 0.78 0.80 0.84 0.86	Categ london chicago losangeles sydney melbourne newyork sanfrancisco toronto	0.57 0.58 0.63 0.70 0.73 0.77 0.79 0.80	Categ chicago london losangeles melbourne sydney newyork toronto sanfrancisco	0.58 0.58 0.65 0.69 0.71 0.77 0.81 0.83	
	 singapore 	0.00	singapore	0.87	singapore	0.89	

Results – in-depth analysis – 300 from test

chicago

london

losangeles

melbourne

sanfrancisco

singapore

sydney

toronto

miami

newyork - 110



Experiment 1:



SELECTED Confusion matrix, without normalization

71 16

1 3

0 0

True labe

22 29 2 12 0 7 4 0

5 27 0

1 11

32 63 19

RANDOM Confusion matrix, without normalization

Experiment 2:

Predicted label	chicago -	187	10	42	20	0	24	11	11	4	21
	london -	2	141	13	52	0	70	26	20	18	30
	losangeles -	3	11	85	18	0	8	26	17	20	18
	melbourne -	8	40	29	102	0	24	31	34	48	47
	miami -	9	12	25	15	0	8	15	21	18	20
	newyork -	64	30	38	33	0	102	28	6	8	43
	sanfrancisco -	4	4	13	20	0	16	118	10	13	4
	singapore -	8	22	22	11	0	24	14	159	19	19
	sydney -	8	12	23	20	0	8	21	9	143	18
	toronto -	7	18	10	9	0	16	10	13	9	80
	81	(a. 10	10530	aet nellos	SULL C	HO. PE	Antran	cis gno	apor si	dur P	01.
						Irue	label				
		SEL	ECTE	D Con	fusio	n mat	label trix, w	vithou	it nor	maliza	ation
	chicago	SELI - 307	CTEI	D Con	ifusio 3	n mat 0	label trix, w 0	vithou 0	t nor	maliza 1	ation 0
	chicago Iondon	SELI 307 0	0 300	D Con 1 0	ifusio 3 0	n mat 0 0	label trix, w 0 0	vithou 0 0	o 0	maliza 1 0	o 0 0
	chicago Iondon Iosangeles	SELI 307 0 5	0 300 21	D Con 1 0 182	ifusion 3 0 15	n mat 0 0 0	label trix, w 0 0 1	vithou 0 0 41	0 0 2	maliza 1 0 29	o 0 1
e	chicago london losangeles melbourne	SELI - 307 - 0 - 5 - 18	0 300 21 29	D Con 1 0 182 16	fusion 3 0 15 203	n mat 0 0 0	label trix, w 0 0 1 2	vithou 0 41 2	0 0 2 1	maliza 1 0 29 18	ation 0 1 12
ed label	chicago Iondon Iosangeles melbourne miami	SELI 307 0 5 18 42	0 300 21 29 61	D Con 1 0 182 16 56	15 203 13	n mat 0 0 0 0	label trix, w 0 1 2 4	vithou 0 0 41 2 9	0 0 2 1 29	malizz 1 0 29 18 49	0 0 1 12 15
Predicted label	chicago Iondon Iosangeles melbourne miami newyork	SEL 307 0 5 18 42 116	0 300 21 29 61 59	D Con 1 0 182 16 56 8	15 203 20	1 rue 0 0 0 0 0 0	label trix, w 0 1 2 4 59	vithou 0 41 2 9 15	0 0 2 1 29 1	maliza 1 0 29 18 49 4	0 0 1 12 15 18
Predicted label	chicago london losangeles melbourne miami newyork sanfrancisco	SELI 307 5 18 42 116 1	0 300 21 29 61 59 8	1 0 182 16 56 8 5	15 203 13 20 2	1 rue 0 0 0 0 0 0 0	label trix, w 0 1 2 4 59 5	vithou 0 41 2 9 15 276	0 0 2 1 29 1 0	maliza 1 0 29 18 49 4 6	ation 0 1 12 15 18 1
Predicted label	chicago london losangeles melbourne miami newyork sanfrancisco singapore	SELI - 307 - 0 - 5 - 18 - 42 - 116 - 1 - 8	CTEI 0 300 21 29 61 59 8 30	D Con 1 0 182 16 56 8 5 15	ifusion 3 0 15 203 13 20 2 2 4	n mat 0 0 0 0 0 0 0 0 0 0	label trix, w 0 0 1 2 4 59 5 1	vithou 0 41 2 9 15 276 4	t norr 0 2 1 29 1 0 224	maliza 1 0 29 18 49 4 6 15	ation 0 1 12 15 18 1 3
Predicted label	chicago london losangeles melbourne miami newyork sanfrancisco singapore sydney	SELL 307 - 0 - 5 - 18 - 42 - 116 - 1 - 8 - 0	0 300 21 29 61 59 8 30 1	Con 1 182 16 56 8 5 15 0	Is 15 203 13 20 2 4 3	n mat 0 0 0 0 0 0 0 0 0 0	label trix, w 0 1 2 4 59 5 1 0	vithou 0 41 2 9 15 276 4 0	t norr 0 2 1 29 1 0 224 0	maliza 1 29 18 49 4 6 15 303	ation 0 1 12 15 18 1 3 2
Predicted label	chicago london losangeles melbourne miami newyork sanfrancisco singapore sydney toronto	SELU - 307 - 0 - 5 - 18 - 42 - 116 - 1 - 8 - 0 - 32	0 300 21 29 61 59 8 30 1 16	Con 1 182 16 56 8 5 15 0 7	In the second se	n mat 0 0 0 0 0 0 0 0 0 0 0 0 0 0	label trix, w 0 1 2 4 59 5 1 0 5	vithou 0 41 2 9 15 276 4 0 2	1 1 0 0 0 2 1 1 0 0 0 0 1 1 0 0 0 0 0 0	maliza 1 0 29 18 49 4 6 15 303 6	ation 0 1 12 15 18 1 3 2 212

True label

Experiment 3:



Selected 300 from test images:



Instacities dataset – ResNet

Fine tuning:



ResNet was able to achieve similar level of accuracy in a shorter time span. It did not require a 5 stage training process (problem solving layers training x3, fine tuning x2).

Results – in-depth analysis

	Experiment	<u>: 1:</u>	Experime	<u>nt 2:</u>	<u>Experim</u>	ent 3:	
10 x Randomly	x Randomly		Test Random:		Test Random:		
	0.4 (+/-0.0242)		0.41 (+/-0.0241)		0.4 (+/-0.0236)	atogony MEAN.	
selected 300	lest Random by catego	Dry MEAN:	Test Random by C	ategory MEAN:	chicogo	a da	
	chicago 0.50		london	0.50	london	0.4/	
from test	Iondon 0.40		locangoloc	0.39	locongoloc	0.35	
	Tosangeles 0.27		rolbourno	0.50	melhounne	0.25	
images:	- merbourne 0.34		neuvenk	0.34	nouvonk	0.25	
0	newyork 0.38		confroncisco	0.39	sanfnancisco	0.41	
	sannancisco 0.48		sannancisco	0.44	singapone	0.40	
	singapore 0.48		sydnov	0.52	sydnay	A 58	
	toponto 0.34		toronto	0.52	toronto	0.36	
	dtype: tloat64		dtype: tloat64	0.23	dtype: tloat64	0.20	
	Test Bandom by catego	NDV STD:	Test Bandom by c	ategory STD:	Test Random by (ategory STD:	
	- chicago 0.027	a	chicago	A A275	chicago	A A266	
	london 0.027	1	london	A A293	london	A A250	
		2	locangeles	A A222	locangeles	0.0255	
	melhouppe 0.025	1	malhourna	0.0222	melhourne	0.0265	
	newyork 0.024	4	newvork	0.0241 0.0131	newvork	0.0126	
	sanfrancisco 0.017	14 16	sanfrancisco	0.0151	sanfrancisco	0.0284	
	singanore 0.02/	0 0	singanore	0.0205	singanore	0.0239	
	sydney 0.024	-0 '9	sydney	0.02/7 0.0265	sydney	0.0255	
	toronto 0.021	5	toronto	0.0205	toronto	0.0226	
	dtype: float64		dtype: float64	0.0210	dtype: float64		
	Test Selected:		Test Selected:		Test Selected:		
Calastad 200	Accuracy 0.68		Accuracy 0.68		Accuracy 0.67	,	
<u>Selected 300</u>	dtype: float64		dtype: float64	, ,	dtvpe: float64		
from toot	Test Selected by cate	gory:	Test Selected by	category:	Test Selected by	category:	
from test	Accurac	V	Ac	curacy	Ac	curacy	
imagas	Categ	, ,	Categ		Categ		
images:	chicago 0.5	3	london	0.54	chicago	0.51	
	london 0.5	6	chicago	0.57	london	0.53	
	svdnev 0.7	0	losangeles	0.67	losangeles	0.65	
	losangeles 0.7	0	newvork	0.69	newyork	0.71	
	melbourne 0.7	3	svdnev	0.70	sydney	0.71	
	newyork 0.7	'5	melbourne	0.75	melbourne	0.74	
	toronto 0.7	'9	toronto	0.78	toronto	0.81	
	sanfrancisco 0.8	0	sanfrancisco	0.81	sanfrancisco	0.81	
	singapore 0.8	7	singapore	0.88	singapore	0.89	

Results – in-depth analysis



THE END!