

Menschen Lernen Maschinelles Lernen

Image-Text Matching: **PicMe**

Seamless Analytics GmbH

Über das Projekt

- Automatische Verknüpfung von Bild- und Textmaterial
 - Es sollen auf Basis eines textuellen Inputs passende Bilder in einer Ordnerstruktur (Dateisystem) gefunden werden

ID	path	text_word	class	conf
1	/email1.txt	cabrio	car	0.9
2	/email1.txt	bulldog	dog	0.7
3	/email1.txt	bulldog	animal	0.9
4	/email2.txt	pitbull	dog	0.8



ID	path	class	conf
1	/bulldog.jpg	dog	0.9
2	/cabrio.jpg	car	0.8
3	/IMGo1.jpg	food	0.7
4	/blume.jpg	flower	0.9

Projektstand Meilenstein-Meeting

- Bildklassifizierung mit Modellen, die über ImageNet trainiert wurden
 - Google Colab (Online Python Notebooks)
 - Download von Bildern
 - Klassifizierung der Bilder
 - Speicherung der Ergebnisse in DB
 - ImageNet Baum-Suche

Zweite Projekthälfte

- **Erweiterung der Bildklassifizierung**
 - Evaluation weiterer Modelle
 - Manueller Modellvergleich → Gewichtung der Modelle
- **Text Stemming von Emails**
 - Download von .eml Files
 - Stemming von Betreff und Email-Body (NLTK Bibliothek)
 - Schreiben der Ergebnisse in eine separate DB-Tabelle
- **Finden von Bildern und Emails über Suchbegriff**
 - Automatisches Zusammenführen von Bildern und Mail-Inhalt

Modellvergleich

- Aufteilung der Bilder auf die Projektmitglieder
 - Jeder für ¼ der Bilder und jeweils alle 8 Netze zuständig
 - Top-1 und Top-5 Accuracy festlegen (Klasse stimmt / stimmt nicht)

	A	B	C	D	E	F	G
1		1	2	3	4	5	6
2		D	D	D	D	D	D
3	URL	http://comprabien	https://upload.	https://www.fazema	https://cdn.prod-carehub	http://4.bp.blog	https://upload
4	MobileNetV2	Laptop.jpg	SteamLocom	xtc.jpg	heroin.jpg	Avocados.jpg	pig.jpg
5	1.Platz	laptop	steam_locomoti	rubber_eraser	syringe	zucchini	wild_boar
6	2.Platz	notebook	projectile	pick	trimaran	banana	warthog
7	3.Platz	space_bar	missile	ocarina	mousetrap	cucumber	hog
8	4.Platz	hand-held_comput	thresher	fig	paddle	orange	hyena
9	5.Platz	desktop_computer	cannon	buckle	corkscrew	grocery_store	red_wolf
10							
11	Top-1 Accuracy (1 / 0)	1	1	0	1	0	1
12	Top-5 Accuracy (1 / 0)	1	1	0	1	0	1
13							
14							
15							

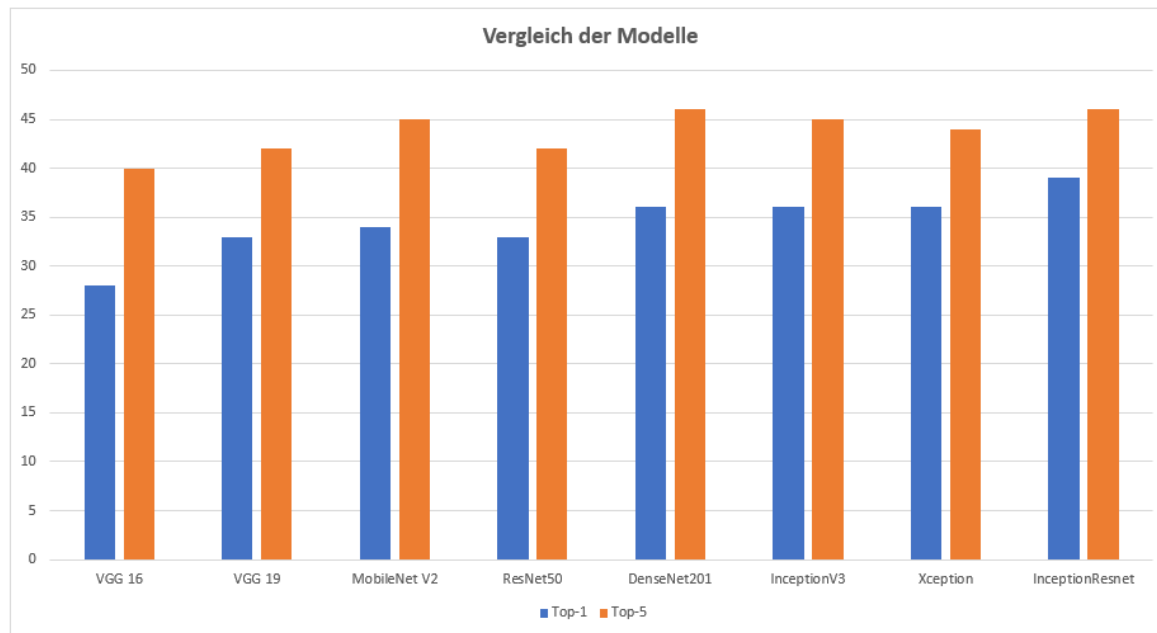
MobileNet V2
ResNet 50
DenseNet 201
Inception V3
Xception
InceptionResnet
Modellvergli ...

Modellvergleich

- Gewichtung festlegen (im Laufe des Projekts verworfen)
 - Aufsummieren der Accuracies → Bestimmung des „besten“ Modells


Gewichtung

	VGG 16	VGG 19	MobileNet V2	ResNet50	DenseNet201	InceptionV3	Xception	InceptionResnet
Top-1	0,394366197		46,5%	47,9%	46,5%	50,7%	50,7%	54,9%
Top-5	0,563380282		59,2%	63,4%	59,2%	64,8%	63,4%	62,0%



Ergebnisspeicherung und Suche

ID	Path	From	To	Subject	Body
1	/tmp/0/07/email1.eml	ml2-sender@web.de	ml2-receiver@web.de	Shopping List	Hi, please don't forget to buy ...



ID	EmailID	StemmingWord
1	1	shop
2	1	butter

ID	ImageID	StemmingWord
1	1	shop
2	2	bird



ID	Path	Model	PredictionClass	Probability
1	/tmp/1/12/image.jpg	VGG16	shop	0.827

Ergebnisspeicherung und Suche

(1) Finde Bilder anhand eines Suchworts:

```
SELECT DISTINCT *
FROM image_list
WHERE PredictionClass = <Suchwort>;
```

(2) Finde Emails anhand eines Suchworts:

```
SELECT DISTINCT a.*
FROM email_list a, email_stemming b
WHERE a.ID = b.emailID
AND b.StemmingWord = <Stem(Suchwort)>;
```

(3) Finde Bild-Email Beziehungen:

```
SELECT DISTINCT a.*, b.*
FROM image_list a, email_list b, image_stemming c, email_stemming d
WHERE image_list.ID = image_stemming.imageID
AND email_list.ID = email_stemming.EmailID
AND image_stemming.StemmingWord = email_stemming.StemmingWord;
```

ID	Path	From	To	Subject	Body
1	/tmp/0/07/email1.eml	ml2-sender@web.de	ml2-receiver@web.de	Shopping List	Hi, please don't forget to buy ...

ID	EmailID	StemmingWord
1	1	shop
2	1	butter

ID	ImageID	StemmingWord
1	1	shop
2	2	bird

ID	Path	Model	PredictionClass	Probability
1	/tmp/1/12/image.jpg	VGG16	shop	0.827

Finaler Stand

=====

Das Stemming-Wort **shop** (abgeleitet vom Wort **shop**) matcht für folgende Image / Mail Kombination:

* From: ML2 Sender <ml2-sender@web.de>
* To: ml2-receiver@web.de
* Subject: Shopping List - Butter, Cheese and Bananas
* Body: Hi,

please don't forget to buy the things. Thank you :-)

Regards

/tmp/image_classification/1/18/PetrolStation.jpg



Aktueller Stand

- **Bildklassifizierung**
 - 71 zusammengetragene Bilder
 - 8 Netze, deren Top-5 Vorhersagen verwertet werden
 - $71 \times 8 \times 5 = 2.840$ Ergebniszeilen in DB
 - Erweiterung durch ImageNet Tree Search auf rund 10.000 Einträge
- **Text-Stemming (NLTK)**
 - Untersuchung von .eml Files (exportierte Emails)
 - Speicherung der Stemming-Wörter in DB
- **Automatischer Abgleich von Bildern und Text**
 - Finden von Bildern und Emails mit passendem Inhalt

Mögliche Erweiterungen

- Evaluierung: Bildklassifizierung vs. Objekterkennung
 - Größerer Suchraum
- Größere Test-Datenbasis schaffen
 - Bing-Image Search (REST API auf RapidAPI)
- Neue DB-Struktur
- Alternative Text-Stemming Algorithmen prüfen

Erfahrungen

- Google Colab: Cloudinstanzen mit Python Notebooks
- Python Programmierung allgemein
- ML-Modelle zur Bildklassifizierung
 - Inkl. Bildvorbereitung
 - Verschiedene Datensätze und Modelle
- Text Stemming mit Natural Language Toolkit

Arbeitsaufwand: ca. 30 Stunden pro Person ~120 Stunden insgesamt

Github Repository (<https://github.com/ml2-picme/PicMe>)