



ICT-TEX course on Digital skills

Topic 7: Introduction to Artificial Intelligence and Machine Learning

The course is developed under Erasmus+ Program Key Action 2: Cooperation for innovation and the exchange of good practices Knowledge Alliance

ICT IN TEXTILE AND CLOTHING HIGHER EDUCATION AND BUSINESS

Project Nr. 612248-EPP-1-2019-1-BG-EPPKA2-KA

The information and views set out in this publication are those of the authors and do not necessarily reflect the official opinion of the European Union. Neither the European Union institutions and bodies nor any person acting on their behalf may be held responsible for the use which may be made of the information contained therein.





Hands-on exercise

CASE STUDY: FABRIC STAIN DEFECT CLASSIFICATION





These slides are part of the Topic 7 on *"Introduction to Artificial Intelligence and Machine Learning"* of the course on Digital skills in Textile and clothing industry.

Check also the main presentation in this topic, as well as the additional reading resources, available in the ICT-TEX platform.





Problem

- In textile industry one of the most common important tasks is quality control and monitoring of fabric production
- Several problems for fabric damage can occur like stains, holes, pattern damage, etc.
- Manual inspection is almost impossible due to the huge endless area that should be inspected
- Al technologies can help for this task for automatic optic inspection of fabric production, by training ML model with different categories of images to identify problems





Problem

- In textile industry one of the most common application of AI is monitoring of fabric production and detection of different defects, like stains, holes, texture, pattern print, etc.
- In this case study we will investigate methods for binary classification, i.e. classification with two classes:
 - Fabrics without defects and
 - Fabrics with different stains.





Challenges

- There is a huge variety of characteristics of stains:
 - size, location, type, shape, number, etc.
- In case the fabric has some printed pattern the task of stain detection is quite challenging. Thus, for simplicity we will consider only plain fabrics (without pattern).
- For additional simplification and generalization we will not consider the color of fabric, i.e. we will use grayscale images





Data

- We need dataset that contain enough classified images of:
 - Fabrics without defects
 - Fabrics with stains
- There is a huge variety of characteristics of stains:
 - size, location, type, shape, number, etc.
- In addition, in case the fabric has some printed pattern the task of stain detection is quite challenging



Data

- Searching for some open data sets we identify the "Fabric stain dataset" in Kaggle that perfectly fits the problem needs.
- Please, download the dataset locally on your computer.

SOFIA UNIVERSITY "ST. KLIMENT OHRIDSKI	ICT-T Ĭ X
---	------------------

k FABRIC	STAIN DATASET Kaggle × +	
	C û ⓓ https://www.kaggle.com	priemshpathirana/fabric-stain-dataset 🛛 🗊 🕁 ± 🗈 🗋 🚳 👁 🔶
≡	kaggle	Q Search
Ø	Home	
Φ	Compete	Dataset
	Data	FABRIC STAIN DATASET
<>	Notebooks	Ink stain, oil stain and dirt stain data set of uniform textured fabric
	Communities	Primesh Pathirana • updated 7 months ago (Version 1)
ଡ	Courses	Data Tasks Notebooks (2) Discussion Activity Metadata Download (415 MB) New Notebook :
\sim	More	
Rece	ently Viewed FABRIC STAIN DATASET	Usability 6.9 License CCO: Public Domain Tags business, arts and entertainment, clothing and accessories
	fabric defect dataset Textile Defect Detection	Description
	African Fabric Images	Context
TA1	Women's clothes	Fabric stain data set has been built for fabric stain defect classification in textile quality control. The data set was built as a part of the fabric defect detection project of the Intellisense Lab of University of Moratuwa, Sri Lanka.
Ē	View Active Events	Content

https://www.kaggle.com/priemshpathirana/fabric-stain-dataset















Different types of stains

Introduction to Artificial Intelligence and Machine Learning - Case study

ICT-TEX course on Digital skills



- The dataset contains:
 - 68 images: "defect free"
 - 398 images: "stain"





Introduction to Artificial Intelligence and Machine Learning - Case study



Framework

- For processing data, we will use Orange Data Mining Framework – free software that provides the basic AI tools in user friendly format that is appropriate for use even from non-technical users.
- Please, install Orange (<u>https://orangedatamining.com/</u>)







Setup

- The core version of Orange contains basic functionalities only. You need to install some add-ons for Image processing.
- To install Image Analytics, please select from the menu Options->Add-ons
- After installation In the Toolbox will appear Image Analytics panel







- Step 1
- Create "New Project"
- Select from "Image Analytics" toolkit the widget "Import Images"
- Set the source folder to be the one that contains your datasets



Import Images

🥶 * File Edit View Widget Options Help			
Learning Analysis Distance MDS Save Self-Organ Transform		Select Top Level Directory $\leftarrow \rightarrow \checkmark \uparrow \models \ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \$	× ひ P Search images
* Prototypes	Import Images ? ×	Organize • New folder	III - 🥐
Create Table Explain Model Explain Prediction Significant Groups Image: Create Table Image: Create Display Image: Create Prediction Image: Create Groups Image: Create Table Image: Create Display Image: Create Frame Image: Create Hub Image: Create Image: Create Hub Image: Create Image: Create Image: Create Frame Image: Create Trape Image: Create Hub Image: Create Frame Image: Create Frame Image: Create Frame	Import Images	 OneDrive This PC 3D Objects Desktop Downloads Music Pictures Videos Windows (C:) 	Date modified 2/20/2021 7:46 PM 2/20/2021 7:46 PM
Save mages Textable Text Field Text Files URLs Preprocess Control Text Files URLs Preprocess Text Field Text Files Preprocess Text Field Text F		Folder: images	Select Folder Cancel

For more information about this widget: <u>https://orangedatamining.com/widget-catalog/image-analytics/importimages/</u>



To inspect the content of the loaded dataset can be added widget "Image Viewer"



Import Images

Note that all images from the folder "stain" are classified with category "stain", and those from "defect_free" have category "defect_free"

> Introduction to Artificial Intelligence and Machine Learning - Case study







Another view for data inspection that we can be used is "Data table" Import Images widget from panel Data

Da	ta			^
	csv	£		
File	CSV File Import	Datasets	SQL Table	
		i		
Data Table	Paint Data	Data Info	Data Sampler	

		📃 Data Table						
2	Image Viewer	Info 466 instances (no missing data) No features Target with 2 values	origin type	category	image name	image va/Desktop/ICT-T image	size	widt
		5 meta attributes	1	defect_free	1	defect_free\1.jpg	1192903	
			2	defect_free	10	defect_free\10.j	1064462	
Oge -		Variables	3	defect_free	11	defect_free\11.j	920379	
1		Show variable labels (if present)	4	defect_free	12	defect_free\12.j	949568	
		Visualize numeric values	5	defect_free	13	defect_free\13.j	631460	
		Color by instance classes	6	defect_free	14	defect_free\14.j	1038393	
	Data Table		7	defect_free	15	defect_free\15.j	1080699	
		Selection	8	defect_free	16	defect_free\16.j	919566	
		Select full rows	9	defect_free	17	defect_free\17.j	970338	
			10	defect_free	18	defect_free\18.j	799133	
		>	11	defect_free	19	defect_free\19.j	1197344	
			12	defect_free	2	defect_free\2.jpg	1086210	
			13	defect_free	20	defect_free\20.j	1142396	
			14	defect_free	21	defect_free\21.j	1225798	
			15	defect_free	22	defect_free\22.j	1136367	
			16	defect_free	23	defect_free\23.j	1253817	
			17	defect_free	24	defect_free\24.j	801313	
			18	defect_free	25	defect_free\25.j	776956	
			19	defect_free	26	defect_free\26.j	1065485	
			20	defect_free	27	defect_free\27.j	1050781	
		Perfore Original Order	21	defect_free	28	defect_free\28.j	902117	
		Nestore original order	22	defect_free	29	defect_free\29.j	1251948	

 \checkmark

Send Automatically

? 🖹 → 466 🕞

defect_free

defect_free\3.jpg

Introduction to Artificial Intelligence and Machine Learning - Case study

×

Α

height



- Image Embedding uses pretrained Deep Learning models to calculate features vectors for each image.
- There are several available DL models that you can choose ("Embedder").
- Please, not that the majority of them will require availability of Internet connection in order to connect to the server for evaluation.



Image Embedding

🚆 Imag 🗕	- 🗆	\times
Settings		
Image attribute:		\sim
Embedder:	Inception v3	V
Google's Inception v3 n	N Inception v3 SqueezeNet (loc	al)
App	VGG-16 VGG-19 Painters DeepLoc openface	
X	Cancel	
2		

SOFIA UNIVERSITY "ST. KLIMENT OHRIDSKI

For more information about this widget: <u>https://orangedatamining.com/widget-catalog/image-analytics/imageembedding/</u>





- For this case study we will use
 - InceptionV3 is Google's deep neural network for image recognition. It is trained on the ImageNet data set. (<u>http://image-net.org/index</u>)

¹ Szegedy, Christian, et al. "Rethinking the inception architecture for computer vision." *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016. <u>https://arxiv.org/abs/1512.00567</u> IM 🗛 GENET

14,197,122 images, 21841 synsets indexed Explore Download Challenges Publications Updates About

Not logged in. Login | Signup

17

ImageNet is an image database organized according to the WordNet hierarchy (currently only the nouns), in which each node of the hierarchy is depicted by hundreds and thousands of images. Currently we have an average of over five hundred images per node. We hope ImageNet will become a useful resource for researchers, educators, students and all of you who share our passion for pictures. Click here to learn more about ImageNet, Click here to join the ImageNet mailing list.



What do these images have in common? Find out!

Research updates on improving ImageNet data

© 2016 Stanford Vision Lab, Stanford University, Princeton University support@image-net.org Copyright infringement



Import Images



Step 5

 Select Image Grid widget to view
 Image Embeddings

	📸 Image Grid																					1		×
	Image Filename Attribute							• •			14													
	- Image cell fit					Cart .		* *		8	1		2/4							1				
Image Viewer	Crop					14 4		*	1			50.00												
	Grid size					1.0	1	1		1									E.		1			
	Columns: 23			N. N	4.	1		4 2					ele-		all and a	and a		and a				100	1	
Data Table	Set size automatically			and a	1	5 8 5 N	1		-				1992			1.00					THE R		No.	
	Labels			9.4	a Ea		1		10		14					1 age	and and							
Embeddings	(No labels)	-	5		10			1			10	ľ.,					20					Name of		
		the state	-	• •		1.					-							1000 1000		<u> </u>				
Image Embedding Image Grid					æ -			-			100				1		101		Por se					
			-	-			25		f. lo										- 91				<u>S</u>	
	3		7	1	1.	15 Ma		4	•													0.0		
			Ends.								¢.													
				*			4		1			and the		and the		-	COLUMN T							
			19-24		1	1							and the second			and the second		R.V.			N.M.			
			*			1	• •	-						1						1				
		4		1:			23			•				.						1000				
					;; ; ;			-	1	3 San			1.80		and a							and the second		
					•	5 M	8 8			are Start	~		原理許	1										
						1.6	• "	2.34	-				al de		1 th								E S	
							-	4	19 A.	a.						No.						4		
										12	• •	3.18	•											
	Apply automatically										••					State of the	T - Star	14						
	2 B B																							







 Including new Data Table View show the new information that was added to the data in a result of image transformation to vectors generated after features selection in Image Embeddings

	10	delectinee	10	delect_free(foj	144122	1904	1400	0.113427	0.0195139	
	11	defect_free	19	defect_free\19.j	1197344	1984	1488	0.440271	0.0125024	
	12	defect_free	2	defect_free\2.jpg	1086210	1984	1488	0.166418	0	
	13	defect_free	20	defect_free\20.j	1142396	1984	1488	0.182529	0	
	> 14	defect_free	21	defect_free\21.j	1225798	1984	1488	0.144828	0.00335589	
	15	defect_free	22	defect_free\22.j	1136367	1984	1488	0.23002	0	
	16	defect_free	23	defect_free\23.j	1253817	1984	1488	0.0893602	0.00940064	
	17	defect_free	24	defect_free\24.j	801313	1984	1488	0.192749	0.00544682	0
	18	defect_free	25	defect_free\25.j	776956	1984	1488	0.213125	0.00387602	
	19	defect_free	26	defect_free\26.j	1065485	1984	1488	0.103953	0.015988	
	20	defect_free	27	defect_free\27.j	1050781	1984	1488	0.190941	0	
	21	defect_free	28	defect_free\28.j	902117	1984	1488	0.0706239	0.03409	
	22	defect_free	29	defect_free\29.j	1251948	1488	1984	0.0776182	0	
	23	defect_free	3	defect_free\3.jpg	638133	1984	1488	0	0.0042975	
	24	defect_free	30	defect_free\30.j	1136502	1984	1488	0.228346	0	
	25	defect_free	31	defect_free\31.j	1213784	1984	1488	0.13187	0.00901064	
	26	defect_free	32	defect_free\32.j	733058	1984	1488	0.319206	0.0446304	
	27	defect_free	33	defect_free\33.j	769720	1488	1984	0.474398	0	
tore Original Order	28	defect_free	34	defect_free\34.j	1070479	1488	1984	0.103713	0.00309253	
ore original order	29	defect_free	35	defect_free\35.j	1197234	1488	1984	0.289641	0.0161295	
Fred Automotion It.	<									>





- The dataset of images for fabric stains is already prepared for the classification task
- We need to select classification model from the Model panel
- We will compare several ML models:
 - SVM (linear)
 - kNN
 - Random forest
 - (Decision) Tree
 - Naïve Bayes







- According to the recommendation for ML method selection our hypothesis is that the best performance will be achieved by SVM method (see the decision highlighted in red), because:
 - We have more than 50 examples in the dataset (we have 466 images)
 - The problem that we try to solve is to identify the category of the image from the optical scanner of fabric
 - The training dataset contains labeled data with two categories
 - The dataset contains 466 <100K examples







Copyright image - source: https://scikit-learn.org/stable/tutorial/machine_learning_map/index.html

Introduction to Artificial Intelligence and Machine Learning - Case study

ICT-TEX course on Digital skills



 The first ML model that is selected is SVM - Support Vector Machine – Linear kernel

😽 SVM		?	×
Name			
SVM			
SVM Type			
SVM		Cost (C):	1.00 🚖
R	egression loss ep:	silon (ɛ):	0.10 🚖
	Regression	cost (C):	1.00 🔷
	Complexity bo	ound (v):	0.50 🜲
Kernel			
() Linear	Kernel: >	r y	
Polynomial	I		
Sigmoid			
- Opumization P	arameters		
	ance:	0.0	
			100
	Apply Auto	matically	
	нррлу нас	anaucany	
	1		

SVM



23







- For evaluation is used Test and Score widget
- We need to select the training and test set.
- There are the following options:
 - Using the current dataset as training set for the ML model and additional dataset to be used as test set. In our case this is not the option, because we do not have additional data
 - Using the same dataset as training set for the ML model and to test with them. This is not considered as a good approach, because the trained model is bias of the data and there are not used new "unseen" data for testing.





- There are options to split the current dataset on two subsets and to use one part of them as training data and the remaining part as test data, called data sampling:
 - One approach is to use random sampling by specifying the ratio between train:test examples. The most common ratio is 2:1, i.e 66% of the data are used as training and the remaining part as test. In this approach some. Although using different train and test set, the random sample has a huge impact on the result obtained, which may differ significantly from the actual classifications.
 - One of the best methods is to use K-fold cross validation. Selecting in advance some integer K. The dataset is splatted in K parts. The training/testing of the ML model is repeated K times, and for each iteration one of the K subsets is chosen as test, and the remaining K-1 subsets are used as training. Common values for K are 3, 5, 7, 10. For smaller datasets is better to choose higher values of K, because only small part of data will not be used for training and there are lower chances to miss significant information in the ML model training.

3-fold cross validation

iteration 1	1 test	2 train	3 train
iteration 2	1 train	2 test	3 train
iteration 3	1 train	2 train	3 test

25





Sampling Cross validation Number of folds: 10 Stratified Cross validation by feature Random sampling Repeat train/test: 10 Training set size: 66 % Stratified Leave one out Test on train data Test on test data Mediel comparison	Sempling Enclose validation Number of folds: 10 Stratified Me hod Cross validation by feature Me hod Report trainflast: 10 Training set size: 65 % Cross validation Media Training set size: 65 % Test on test data Model comparison Model Comparison Model comparison	Sampling Image: Standing of folds: Standing set stail Image: Standing of folds: Image: Standing of fold: Image: Standing of fold: <	Senging because violation in Results Provided in Prov	🗯 Test and Score	-	
Random sampling Repeat brain/test: 10 Training set size: 66 % Stratified Leave one out Test on train data Test on test data	Random sampling Repest train/test: 10 Training set size: 66 % Stratified Leave one out Test on train data Test on train data Target Cless Model comparison Model comparison Model Comparison Model Comparison	Rendom sampling Repet train/text: 10 Training set size: 66 % Stratified Leave one out Text on text data Model comparison Model comparison Model comparison	Residen sampling Repet train/text: 20 Training set size: 26 56:00 Stabilided Larser one cut Text on text date Text on text date Model comparison Model comparison Model comparison Negligible difference: 0.1	Sampling Cross validation Number of folds: 10 Stratified Cross validation by feature	Evaluation Results	
Test on train data Test on test data Target Class Model comparison	O Test on train data O Test on test data Target Cless Model comparison Model Comparison Model comparison Image: State of the stat	Test on test dets Target Cless Model Comparison Model comparison Model difference:	Test on test data Terget Class Model comparison Model comparison Negligible difference: 0.1	Random sampling Repeat train/test: 10 Training set size: 66 % Stratified Leave one out		
	Negligible difference: 0.1	Negligible difference: 0.1	Madei Lomperson	Test on train data Test on test data Target Class	Model comparison	

26





Evaluation Metrics

$$Precision = \frac{TP}{TP + FP}$$
$$Recall = \frac{TP}{TP + FN}$$

 $F1 = 2 \cdot \frac{Precission \cdot Recall}{Precission + Recall}$





true positive (TP) true negative (TN) false positive (FP) false negative (FN)

Source, License <u>CC BY-SA 4.0</u>, <u>Walber</u>: <u>https://upload.wikimedia.org/wikipedia/commons/thumb/2/26/Precisionrecall.svg/700px-Precisionrecall.svg.png</u>

Introduction to Artificial Intelligence and Machine Learning - Case study





For comparison with other ML models, we add kNN, Random forest, (Decision) Tree and Naïve Bayes widgets from Model panel, using their default parameters







Introduction to Artificial Intelligence and Machine Learning - Case study





Discussion

- The results of ML classification models training show that the best performance have:
 - SVM (Linear) with highest F1 score 0.865
 - Naïve Bayes has highest Precision 0.877, but lacks Recall.
 - SVM (Linear) has highest Recall 0.867
- The huge variety of stains features, like shapes, size, number of stains, does not allow to achieve better performance of ML models due to relatively small training dataset, that does not allow to be investigated the majority of possible features combinations.
- The results of demonstrated ML classification models are promising and show that the task can be solved with satisfactory precision for stain detection in fabric manufacturing.





References

- The material of these slides is based on the following resources:
 - Applications of AI in Textile Industry <u>https://frontier.cool/blogposts/importance-machine-learning-textile-industry</u>
 - Orange widget catalog: https://orangedatamining.com/widget-catalog/
 - Orange Data Mining Framework: <u>https://orangedatamining.com/</u>

CONTACTS

Coordinator: Technical University of Sofia

Project coordinator:

assoc. prof. Angel Terziev, PhD aterziev@tu-sofia.bg

Web-site: ICT-TEX.eu



Author:

Assoc. Professor Svetla Boytcheva Institute of Information and Communication Technologies, Bulgarian Academy of Sciences

KNOWLEDGE ALLIANCE

ICT-T**I**X

ICT IN TEXTILE AND CLOTHING HIGHER EDUCATION AND BUSINESS

These slides and the materials included in these slides (including references) are for educational purposes only. The use of slides should be done with correct citation and only for educational purposes.

The information and views set out in this publication are those of the authors and do not necessarily reflect the official opinion of the European Union. Neither the European Union institutions and bodies nor any person acting on their behalf may be held responsible for the use which may be made of the information contained therein.