Documentation of D4.4: SILKNOW Image Classification

Release 0.0.1

LUH

Dec 06, 2019

CONTENTS

1 Overview of the documentation **1**

2 SILKNOW Image Classification **3**

2.1 silknow_image_classification Toolbox 3

Python Module Index **7**

Index **9**

CHAPTER ONE

OVERVIEW OF THE DOCUMENTATION

This software provides python functions for the classification of images and training and evaluation of classification models. It consists of five main parts: The creation of a dataset, the training of a new classifier, the evaluation of an existing classifier, the classification of images using an existing classifier and the combined training and evaluation in a five-fold cross validation. All functions take configuration files as an input and generally write their results in specified paths. The format required for the configuration files is described in Deliverable D4.4.

The requirements for the silknow_image_classification toolbox are python 3.7.4 and the following python packages:

- opencv-python
- numpy
- urllib3
- pandas
- tqdm
- tensorflow (1.13.1)
- tensorflow-hub (0.6.0)
- matplotlib
- sklearn
- xlrd

CHAPTER

TWO

SILKNOW IMAGE CLASSIFICATION

2.1 silknow_image_classification Toolbox

All functions in the image classification package as well as a short description of them are listed in the following table. Afterwards, the operating principle of each function is explained and the respective input and output parameters are described.

Name of the function	Short description of the function
CreateDataset	Creates a dataset with samples for the training and evaluation (Data Preparation).
train_CNN_Classifier	Trains a new classification model. (Training)
evaluate_CNN_Classifier	Evaluates an existing classification model. (Evaluation)
crossvalidate_CNN_Classifier	Trains and evaluates a new model using cross validation.
apply_CNN_Classifier	Classifies images using an existing classification model. (Classification)

Process flow of the image processing module and its embedding in the context of SILKNOW.

Created on Fri Nov 29 18:37:39 2019

@author: clermont

`silknow_image_classification.`**`apply_CNN_Classifier`**`(`*configfile*`)`

Applies the trained classifier to new data.

Arguments:

configfile (*string*): This variable is a string and contains the name of the configfile. All relevant information for applying the trained classifier is in this file. The configfile has to be stored in the same location as the script executing the classification function.

Returns No returns. The classification result will be written automatically into result file in the master direction. The name of this file can be chosen by the user in the control file.

`silknow_image_classification.`**`createDataset`**`(`*configfile*`)`

Creates the dataset for the CNN.

Arguments:

configfile (*string*): This variable is a string and contains the name of the configfile. All relevant information for the dataset creation are in this file. The configfile has to be stored in the same location as the script executing the classification function.

Returns: No returns. The classification result will be written automatically into result file in the master direction. The name of this file can be chosen by the user in the control file.

`silknow_image_classification.`**`crossvalidate_CNN_Classifier`**`(`*configfile*`)`

Carries out training of the CNN with cross validation.

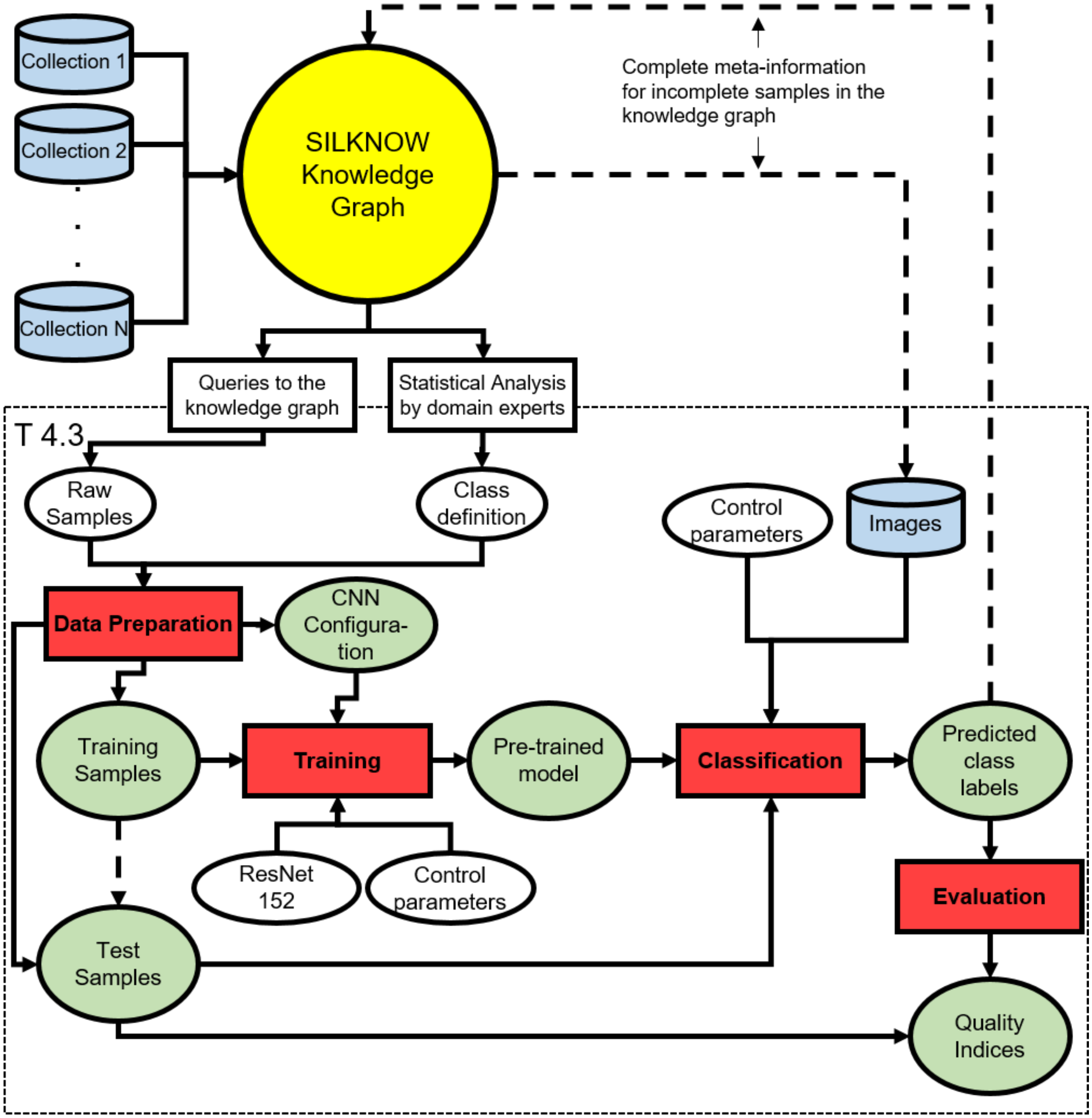

Fig. 1: Process flow of the SILKNOW image classification module and its embedding in the context of SILKNOW.

Arguments:

configfile (*string*): This variable is a string and contains the name of the configfile. All relevant information for the dataset creation are in this file. The configfile has to be stored in the same location as the script executing the classification function.

Returns: No returns. The results of the evaluation are stored automatically in the directory given in the configfile.

`silknow_image_classification.`**`evaluate_CNN_Classifier`**`(`*configfile*`)`

Evaluates a pre-trained CNN.

Arguments:

configfile (*string*): This variable is a string and contains the name of the configfile. All relevant information for the evaluation is in this file. The configfile has to be stored in the same location as the script executing the evaluation function.

Returns: No returns. The results of the evaluation are stored automatically in the directory given in the configfile.

`silknow_image_classification.`**`train_CNN_Classifier`**`(`*configfile*`)`

Trains a classifier based on top of a pre-trained CNN.

Arguments:

configfile (*string*): This variable is a string and contains the name of the configfile. All relevant information for the training is in this file. The configfile has to be stored in the same location as the script executing the training function.

Returns: No returns. The trained graph (containing the tfhub_module and the trained classifier) is stored automatically in the directory given in the control file.

PYTHON MODULE INDEX

s

silknow_image_classification, 3

INDEX

A

apply_CNN_Classifier() (*in module silknow_image_classification*), 3

C

createDataset() (*in module silknow_image_classification*), 3

crossvalidate_CNN_Classifier() (*in module silknow_image_classification*), 3

E

evaluate_CNN_Classifier() (*in module silknow_image_classification*), 5

S

silknow_image_classification (*module*), 3

T

train_CNN_Classifier() (*in module silknow_image_classification*), 5