py_ciu_image

Kary Främling

Feb 16, 2024

CONTENTS:

This package implements the Contextual Importance and Utility (CIU) method.

Classes:

- *CIUimage.CIUimage*: The CIUimage class implements the Contextual Importance and Utility method for explaining image classification.

Example:

```
# Example code using the module
import CIUimage as CIU
out_names = ["NonBleeding", "Bleeding"] # Can also be "None".
ciu_object = CIU.CIUimage(model, out_names)
ciu_sp_result = ciu_object.explain(img_to_xplain)
```

class CIUimage.CIUimage.**CIUimage**(*model*, *out_names=None*, *predict_function=None*, *background_color=(190, 190, 190)*, *strategy='straight'*, *neutralCU=0.5*, *segments=None*, *nbr_segments=50*, *compactness=10*, *debug=False*)

This class implements the Contextual Importance and Utility (CIU) explainable AI method for explaining image classifications.

Parameters

- **model** – ML model to use.
- **out_names** (*list*) – List of output class names to be used.
- **predict.function** – Function that takes a list of images and return a numpy.ndarray with output probabilities. If this is *None*, then it is set to *model.predict_on_batch* by default.
- **background_color** – Background color to use for "transparent", in RGB. In the future this will be modified for supporting more than one different colors, patterns or other perturbation methods.
- **strategy** (*str*) – Defines CIU strategy. Either "straight" or "inverse".
- **neutralCU** – CU value that is considered "neutral" and that provides a limit between negative and positive influence in the "Contextual influence" calculation CIx(CU - neutralCU).
- **segments** – np.array of same dimensions as image, with segment index for every pixel. The default is *None*, which signifies that the default SLIC method will be used for creating superpixels.
- **nbr_segments** (*int*) – The amount of target segments to be used by the SLIC algorithm.
- **compactness** (*int*) – The compactness of the segments accounting for proximity or RGB values. The default is 10 and logarithmic.
- **debug** (*bool*) – Display variable values and messages for debugging purposes.

explain(*image*, *strategy=None*)

Calculate CIU values for the given image.

Parameters

- **image** – Image object to explain.
- **strategy** (*str*) – Defines CIU strategy. Either "straight" or "inverse". The default is "None", which causes self.strategy (from constructor) to be used instead.

image_influential_segments_only(*ind_output=0*, *Cinfl_limit=None*, *type='why'*, *CI_limit=0.5*, *CU_limit=0.51*)

Method that returns a version of the explained image that shows only superpixels above (for *type="why"*) or below (for *type="whynot"*) threshold values. Threshold values can be either on Contextual influence (*Cinfl_limit*) OR a combination of CI values equal or over *CI_limit* and CU values equal or over/under (depending on *type*) *CU_limit*.

REMARK: Check proper operation with CI and CU; meanwhile, it's recommended to use *Cinfl_limit* instead.

Parameters

- **ind_output** (*int*) – Index of output class to use.
- **Cinfl_limit** (*float*) – Contextual influence limit to use. If *None*, then use CI&CU limits instead.
- **type** (*str*) – Can take values "why" or "whynot".
- **CI_limit** (*float*) – Inclusion limit (>=) for Contextual Importance (CI).
- **CU_limit** (*float*) – Inclusion limit (>=) for Contextual Utility (CU).

CHAPTER

ONE

INDICES AND TABLES

- genindex
- modindex
- search

PYTHON MODULE INDEX

C

INDEX