Short description

Control elements are in accuracy_pp_rules/main.py. Furthermore, accuracy_pp_rules/configuration.py needs to be adjusted. The other libraries work in the background.

Contains three functionalities:

1. Post-Processing real data:
    a. Applies the desired post-processing rules to the .csv files generated by boris_to_csv.py and outputs them into the described structure.
2. Evaluation:
    a. For every night it is possible to compare the outcome of BOVIDS with the manual annotation.
    b. It is furthermore possible to apply post-processing rules to the real data (manually annotated) and compare this ("realPP") with the outcome of BOVIDS.
    c. Finally, those realPP data can be compared with the real data to evaluate how large a systematic error caused by the choice of the post-processing rules is.
3. Overview:
    a. Step 1 produces up to six .xlsx sheets per individual (all three comparisons, total and binary classification task). Each sheet contains the information described in readme.md (paragraph "evaluation") per night. The second steps provides a method to generate an overview about all individuals and all comparisons.

Requirements

- packages: numpy, openpyxl, scikit-learn, scipy, pandas
- BOVIDS specific packages: complete folder /accuracy_pp_rules/

Step 1 – open spyder:

- Terminal / shell:
    - conda activate bovids
    - spyder

Step 2 – adjust parameters:

configuration.py

- POST_PROCESSING_MAPPING_TOTAL = Dictionary such that the keys are either individual codes or species names and the corresponding value denotes the name of the post-processing rule that should be applied to the individual / species. The script will first search for the individual code, if this does not exist, the post-processing rule of the species is applied. [dictionary]
- POST_PROCESSING_MAPPING_BINARY = See above but for the binary classification task. [dictionary]
- VIDEO_START_SPECIAL = If some species_zoo combinations are not recorded from GLOBAL_STARTING_TIME, those can be specified here. [dictionary]
- VIDEO_END_SPECIAL = If some species_zoo combinations are not recorded until GLOBAL_ENDING_TIME, those can be specified here. [dictionary]
- ERROR_TOLERANCE_INTERVALS = The script will only report errors of at least the length (number time-intervals) indicated here. [integer]
- BEHAVIOR_NAMES = List of the names of the behaviors. [list]
- CONF_SPALTEN = List of behaviors appearing as columns of each confusion matrix in the output. [list]
- CONF_ZEILEN = List of behaviors appearing as rows of each confusion matrix in the output. Normally, truncation is not included. Be aware that 0, 1 ,2, 3, 4 (the indices of the list) belong to specific poses estimated by the action classifiers. [list]
- INTERVAL_LENGTH = 7 (seconds) [integer]
- GLOBAL_STARTING_TIME = Global start of the observation time. [integer]
- GLOBAL_ENDING_TIME = Global end of the observation time. [integer]
- GLOBAL_OBSERVATION_HOURS = Number of hours recorded. [integer]
- POST_PROCESSING_RULES = Might be a copy of the corresponding entry of global_configuration.py. Can also be extended by more and different post-processing rules if the script is used to try and evaluate different sets of rules. [dictionary]

main.py

- INDIVIDUALCODES_BOVIDS : List of individuals for which the real data (.csv from BORIS) should be converted into post-processed data in BOVIDS' structure.
- CSV_FOLDER = Path (folder) to DATA_STORAGE/Auswertung/ (see readme.md, paragraph "action classification storage") [string]

- E.g.: …DATA_STORAGE/Auswertung/
- OUTPUT_FOLDER_BOVIDS = Path (folder) to the desired output. Normally, this should coincide with DATA_STORAGE/Auswertung/ (see readme.md, paragraph “action classification storage”) [string]
- E.g.: …DATA_STORAGE/Auswertung/
- USE_SPECIES: If set to True, the comparison will be conducted for all available data belonging to individuals of the species’ given by SPEC_ACCURACY. If False, only those individuals of INDIVIDUAL_CODE_ACCURACY will be evaluated. [boolean]
- SPEC_ACCURACY = List of species names that will be evaluated if USE_SPECIES is True. [list]
- INDIVIDUAL_CODE_ACCURACY = List of individual codes that will be evaluated if USE_SPECIES is False. [list]
- BASE_FOLDER_CSV = Path (folder) to DATA_STORAGE/Auswertung/ (see readme.md, paragraph “action classification storage”) [string]
    - E.g.: …DATA_STORAGE/Auswertung/
- BASE_FOLDER_AI_PREDICTION = Path (folder) to the storage of the prediction by BOVIDS (FINAL_STORAGE_PREDICTION_FILES). [string]
- OUTPUT_FOLDER_ACCURACY = Destination (folder). For each individual, the script will create one subfolder containing the described .xlsx files to evaluate accurary. [string]
- COMPARE_REAL_WITH_AI: If True, the comparison between the output of BOVIDS and the real data is conducted. If True, BASE_FOLDER_CSV and BASE_FOLDER_AI_PREDICTION need to exist. If False, the comparison is skipped. [boolean]
- COMPARE_REAL_WITH_REALPP: If True, the real data will be compared to the real data with post-processing applied (realPP). Only BASE_FOLDER_CSV is required. If False, this comparison is skipped. [boolean]
- COMPARE_REAL_PP_WITH_AI: If True, the outcome of BOVIDS is compared to realPP. If False, this is skipped. Requires BASE_FOLDER_CSV and BASE_FOLDER_AI_PREDICTION. [boolean]
- TOTAL: If True, the accuracy of the total classification task is reported. [boolean]
- BINARY: If True, the accuracy of the binary classification task is reported. [boolean]

Parameters to obtain the overview sheets:

- INPUT_FOLDER_FOR_OVERVIEW = Corresponds to OUTPUT_FOLDER_ACCURACY. [string]
- OUTPUT_FOLDER_OVERVIEW = Destination (path) for the overview sheets. [string]

Step 3 – run the script:

- Save configuration.py and run main.py.
- To conduct post-processing, enter: *generate_bovids_files()*
- To generate all files of all comparisons, type: *generate_accuracy()*
- Given all those files, type *generate overview()* to generate the overview files.