gulp-metadata-modifier

Release 0.0.4

Will Price

Feb 06, 2020

CONTENTS

Update the metadata in a gulp directory from a given dataframe

```
usage: sphinx-build [-h] [--drop-unknown] [--ignore-missing-examples]
                    [--disable-backup]
                    annotations_pkl GULP_DIR [GULP_DIR ...]
```

CHAPTER

ONE

POSITIONAL ARGUMENTS

annotations_pkl Path to pickled DataFrame containing all annotations. The index must correspond to the ID of the gulped examples. All the columns' values will be copied into the first item of the metadata field for that example.

GULP_DIR Gulp directories to update

CHAPTER

TWO

NAMED ARGUMENTS

--drop-unknown Drop entries for examples not found in annotations DataFrame

Default: False

--ignore-missing-examples If an example is missing from annotations_pkl then this script will raise an error unless –drop-unknown is set, however if you simply want to leave that metadata untouched when –drop-unkonwn is not set, then set this flag.

Default: False

--disable-backup Disable creation of gmeta_X.bak files when updating metadata

Default: True

gulp_metadata_modifier.**modify_metadata**(*gulp_dir*, *transform_func*, *, *drop_nones=False*, *backup=True*)

Update the metadata in `gulp_dir` according to the user provided function `transform_func` which takes in a single example's id and metadata and transforms it.

Parameters

- **gulp_dir** (`Union[Path, str]`) – Gulp directory containing `.gmeta` and `.gulp` files.
- **transform_func** (`Callable[[str, Dict[str, Union[List[List[int]], List[Dict[str, Any]]]]], Optional[Dict[str, Union[List[List[int]], List[Dict[str, Any]]]]]]`) – User provided function to transform the metadata of an example in some way. This should take in the example id and old metadata and return either `None` if the segment is to be dropped (if `drop_nones=True` or to be left unchanged otherwise) or the updated metadata dict.
- **drop_nones** (`bool`) – If set and `transform_func` returns None, then remove the segment from the gulp meta dict.
- **backup** (`bool`) – Make `.bak` files for all `.gmeta` files.

Return type `None`

gulp_metadata_modifier.**modify_all_gulp_dirs**(*gulp_dir_root*, *transform_func*, *gulp_dir_pattern=re.compile('.*gulp.*')*)

Apply `transform_func` to all gulp metadata within all gulp directories matching `gulp_dir_pattern` in `gulp_dir_root`.

Parameters

- **gulp_dir_root** (`Path`) – Root directory to search for gulp directories
- **transform_func** (`Callable[[str, Dict[str, Union[List[List[int]], List[Dict[str, Any]]]]], Dict[str, Union[List[List[int]], List[Dict[str, Any]]]]`) – User provided function to transform an example's metadata

- **gulp_dir_pattern** (Pattern) – A directory below gulp_dir_root is considered a gulp directory if it matches this pattern.

See also:

modify_metadata()

Return type None

gulp_metadata_modifier.**GulpExampleId**
alias of builtins.str

PYTHON MODULE INDEX

INDEX

G

M