Optilearn.text.EmoTextHandler

08 November 2024   19:11

EmoTextHandler

The EmoTextHandler function is designed to manage emojis and emoticons in text, allowing you to remove, replace, add, or customize the behavior of emojis and emoticons based on your needs. This function comes with predefined sets of emojis and emoticons, which you can modify through various parameters. Here's a breakdown of its functionality:

Key Features:
- **Processing Selection**: Decide whether to process only emojis (emj), only emoticons (emt), or both (both).
- **Action Type**: Choose to either remove (remove) or replace (replace) emojis or emoticons in the text.
- **Custom Additions**: Add new emojis or emoticons with meanings via a dictionary, allowing the function to process custom symbols beyond the predefined set.
- **Custom Removals**: Remove specific emojis or emoticons from the predefined sets to avoid their processing or removal.
- **Batch Removal**: Optionally, remove all predefined emojis, emoticons, or both to work with only specific additions or for cases requiring strict customization.
- **Case Modification**: Control the letter case of the output text, whether to retain original case, convert all to lowercase, or all to uppercase.
- **Flexible Filtering**: Remove any words, emojis, or emoticons provided in a list via remove_from_list, offering versatile control over text content.

Parameters Overview:
- **text**: The main text input to be processed.
- **process_on**: Specifies whether to focus on emojis, emoticons, or both.
- **process_type**: Determines the action to take with the emojis or emoticons (remove or replace).
- **add** and **add_em**: Specify to add custom emojis or emoticons. Choose which to add using add, then provide the symbols and meanings in add_em.
- **remove** and **remove_em**: Indicate which to remove, either from the predefined emoji or emoticon sets. remove_em provides the specific symbols to exclude.
- **remove_all**: Clears predefined emojis, emoticons, or both, ideal when needing a highly customized set.
- **case**: Sets the case for the output text (same, lower, or upper).
- **remove_from_list**: Accepts a list of specific words, emojis, or emoticons to filter out directly from the text.

Returns:
- **Processed Text**: The function returns the modified text, processed according to the specified parameters.

Usage Examples:
1. **Replacing Emojis**:

```
processed_text = EmoTextHandler(text=text, process_on='emj', process_type='replace',
print(processed_text)
```

2. **Adding and Removing Specific Emojis/Emoticons**:

```
processed_text = EmoTextHandler(
    text=text,
    remove_all='emt',
    add='emt',
    add_em={"( ͡° ͜ʖ ͡° )": "Lenny Face with Glasses"},
    remove='emj',
    remove_em=["👿", "😁"]
)
print(processed_text)
```

Notes:
- Offers a flexible way to customize emoji and emoticon handling in text.
- Provides control to include or exclude specific symbols, letting you make it work for personalized sets.
- Allows broad or narrow removal and replacement actions based on your specific requirements.

This function is especially useful in text processing tasks where emojis and emoticons need to be managed systematically, such as in social media analysis, chat-based applications, and NLP projects.