[Code] Text Preprocessing & Tokenize

```
import os
import re
import pandas as pd
import numpy as np
import json
from tqdm.auto import tqdm
from pathlib import Path
from dask import dataframe as dd
from dask.diagnostics import ProgressBar

pd.set_option('display.max_columns', None)  # or 1000
pd.set_option('display.max_rows', 1000)  # or 1000
pd.set_option('display.max_colwidth', -1)  # or 199
tqdm.pandas()
ProgressBar().register()

# Your data path (change plz)
DATA_DIR = Path("../SocioGraph/data/Twint_crawing/Hindi/parquet")

# Read data
df = dd.read_parquet(DATA_DIR / "hindi_root_total_no_1469358.parquet").compute()
len(df)

# Basic Cleaning Text Function
## You should remove Special characters (ex. !,.$#%@)

def CleanText(readData):

    # Remove Retweets
    text = re.sub('RT @[\w_]+: ', '', readData)

    # Remove Mentions
    text = re.sub('@[\w_]+', '', text)

    # Remove Hashtag (optional)
    text = re.sub('[#]+[0-9a-zA-Z_]+', ' ', text)

    # Remove or Replace URL
    text = re.sub(r"http[s]?://(?:[a-zA-Z]|[0-9]|[$-_@.&+]|[!*\(\),]|(?:%[0-9a-fA-F][0-9a-fA-F]))+", ' ', text) # start with http ur
    text = re.sub(r"[-a-zA-Z0-9@:%._\+~#=]{1,256}\.[a-zA-Z0-9()]{2,6}\b([-a-zA-Z0-9()@:%_\+.~#?&//=]*)", ' ', text) # not start with
```

```
    # Remove Zero-width non-joiner (optional)
    text = re.sub('\u200c' , ' ', text)

    # Remove Garbage Words (ex. &lt, &gt, etc)
    text = re.sub(r'[^\w\s]',' ',text)

    # Remove English (If you want, activate the code)
    #text = re.sub('[a-zA-Z]' , ' ', text)

    # Remove newline
    text = text.replace('\n',' ')

    # Remove multi spacing & Reform sentence
    text = ' '.join(text.split())

    return text

# Check the code above
## you should adjust the code slightly

SAMPLE_TEXT = df.iloc[0]["tweet"]
print(f"Before cleaning text: {SAMPLE_TEXT}")
print(f"After cleaning text: {CleanText(SAMPLE_TEXT)}")

# Need approperate Tokenizer: you should import
## This code is for Farsi
from parsivar import Tokenizer

# Find the stopword in your language and import it
## This code is for Farsi
farsi_stopwords = pd.read_csv(DATA_DIR / "persian_stopwords.txt", delimiter="\t", names=["stopwords"])

# Tokenize!
my_tokenizer = Tokenizer()

def Preprocessing(readData):

    #### Clean text
    sentence = CleanText(readData)

    #### Tokenize
    morphs = my_tokenizer.tokenize_words(sentence)

    # Remove Stopwords
    morphs[:] = (morph for morph in morphs if morph not in farsi_stopwords["stopwords"].tolist()) ## Farsi case

    # Remove length-1 words
    morphs[:] = (morph for morph in morphs if not (len(morph) == 1))

    # Result pop-up
    result = []
    for morph in morphs:
        result.append(morph)

    return result

# Check the code above
## you should adjust the code slightly

SAMPLE_TEXT = df.iloc[1]["tweet"]
print(f"Before cleaning text: {SAMPLE_TEXT}")
print(f"After cleaning text: {Preprocessing(SAMPLE_TEXT)}")

# Run!
## Total dataframe
df["cleaned"] = df['tweet'].progress_map(CleanText)
df["tokenized"] = df['tweet'].progress_map(Preprocessing)

# Save
# Your path
## Your own file name
df.to_pickle(DATA_DIR / "total_tokens_hindi.pkl", compression='gzip')

# Check the saved file
check_df = pd.read_pickle(DATA_DIR / "total_tokens_hindi.pkl", compression='gzip')
len(check_df)

check_df[["tweet", "cleaned", "tokenized"]].head(30)
```