Reddit DataFrame Schema

Here's the code that I have that will extract various subsets of the Reddit data on the cluster.

Running this will take time, so please avoid running it when the cluster is busy with assignment work. Make sure you have access to the web frontend of your job so you can monitor its progress. It should take 15 minutes or so, as long as you keep the filter on `year` which will eliminate most of the files so they don't have to be read at all.

If you want to do any further analysis of the data on the cluster, extract a subset like this first (call it an "ETL task"). Then work on the subset with subsequent programs.

Of course, comment-out parts you don't need (comments or submissions: as long as you don't write the file, lazy evaluation will prevent the calculation from happening).

```
import sys
from pyspark.sql import SparkSession, functions, types, Row

spark = SparkSession.builder.appName('reddit extracter').getOrCreate()

reddit_submissions_path = '/courses/datasets/reddit_submissions_repartitioned/'
reddit_comments_path = '/courses/datasets/reddit_comments_repartitioned/'
output = 'reddit-subset'

comments_schema = types.StructType([
    types.StructField('archived', types.BooleanType()),
    types.StructField('author', types.StringType()),
    types.StructField('author_flair_css_class', types.StringType()),
    types.StructField('author_flair_text', types.StringType()),
    types.StructField('body', types.StringType()),
    types.StructField('controversiality', types.LongType()),
    types.StructField('created_utc', types.StringType()),
    types.StructField('distinguished', types.StringType()),
    types.StructField('downs', types.LongType()),
    types.StructField('edited', types.StringType()),
    types.StructField('gilded', types.LongType()),
    types.StructField('id', types.StringType()),
    types.StructField('link_id', types.StringType()),
    types.StructField('name', types.StringType()),
    types.StructField('parent_id', types.StringType()),
    types.StructField('retrieved_on', types.LongType()),
    types.StructField('score', types.LongType()),
    types.StructField('score_hidden', types.BooleanType()),
    types.StructField('subreddit', types.StringType()),
    types.StructField('subreddit_id', types.StringType()),
    types.StructField('ups', types.LongType()),
    types.StructField('year', types.IntegerType()),
    types.StructField('month', types.IntegerType()),
])
```

```
submissions_schema = types.StructType([
    types.StructField('archived', types.BooleanType()),
    types.StructField('author', types.StringType()),
    types.StructField('author_flair_css_class', types.StringType()),
    types.StructField('author_flair_text', types.StringType()),
    types.StructField('created', types.LongType()),
    types.StructField('created_utc', types.StringType()),
    types.StructField('distinguished', types.StringType()),
    types.StructField('domain', types.StringType()),
    types.StructField('downs', types.LongType()),
    types.StructField('edited', types.BooleanType()),
    types.StructField('from', types.StringType()),
    types.StructField('from_id', types.StringType()),
    types.StructField('from_kind', types.StringType()),
    types.StructField('gilded', types.LongType()),
    types.StructField('hide_score', types.BooleanType()),
    types.StructField('id', types.StringType()),
    types.StructField('is_self', types.BooleanType()),
    types.StructField('link_flair_css_class', types.StringType()),
    types.StructField('link_flair_text', types.StringType()),
    types.StructField('media', types.StringType()),
    types.StructField('name', types.StringType()),
    types.StructField('num_comments', types.LongType()),
    types.StructField('over_18', types.BooleanType()),
    types.StructField('permalink', types.StringType()),
    types.StructField('quarantine', types.BooleanType()),
    types.StructField('retrieved_on', types.LongType()),
    types.StructField('saved', types.BooleanType()),
    types.StructField('score', types.LongType()),
    types.StructField('secure_media', types.StringType()),
    types.StructField('selftext', types.StringType()),
    types.StructField('stickied', types.BooleanType()),
    types.StructField('subreddit', types.StringType()),
    types.StructField('subreddit_id', types.StringType()),
    types.StructField('thumbnail', types.StringType()),
    types.StructField('title', types.StringType()),
    types.StructField('ups', types.LongType()),
    types.StructField('url', types.StringType()),
    types.StructField('year', types.IntegerType()),
    types.StructField('month', types.IntegerType()),
])

def main():
    reddit_submissions = spark.read.json(reddit_submissions_path, schema=submissions_schema)
    reddit_comments = spark.read.json(reddit_comments_path, schema=comments_schema)

    subs = ['Genealogy', 'xkcd', 'optometry', 'Cameras', 'scala']
    subs = list(map(functions.lit, subs))
```

```
    reddit_submissions.where(reddit_submissions['subreddit'].isin(subs)) \
        .where(reddit_submissions['year'] == 2016) \
        .write.json(output + '/submissions', mode='overwrite', compression='gzip')
    reddit_comments.where(reddit_comments['subreddit'].isin(subs)) \
        .where(reddit_comments['year'] == 2016) \
        .write.json(output + '/comments', mode='overwrite', compression='gzip')


main()
```

From there, you can have a look at the data and make sure everything seems reasonable:

```
hdfs dfs -ls reddit-subset/submissions
hdfs dfs -ls reddit-subset/comments
hdfs dfs -du -s -h reddit-subset/submissions
hdfs dfs -du -s -h reddit-subset/comments
```

Updated Thu Nov. 02 2023, 14:24 by ggbaker.