Algorithm Design 21/22

Hands On 4 - Tweets

Federico Ramacciotti

1 Problem

Questions:

1. Count the percentage of happy users in the different moments of the day (morning, afternoon, evening, night).
    - Discuss what you find if compute also the percentage of unhappy users. Do the two percentages sum to 100%? Why?
2. Spell the 30 favorite words of happy users
3. Find the number of distinct words used by happy users
    - How could we exclude words repeated only once?
4. Decide if in general happy messages are longer or shorter than unhappy messages

https://replit.com/@geraci/esercizioTweets

2 Solution

2.1 Question 1

The best way to count the percentage of happy users in different moments of the day is to use linear counters: in this case we use 6 linear counters, 4 for the moments of the day and 2 for the mood of the user. In this way we can know if a certain user has a certain property. In order to know the amount of users that have more than one property, we just have to do the logical AND between them and count the results that are True. If a user makes more than one tweet per mood (i.e. a user makes happy and unhappy tweets), it will count as more than one user, because it will be set to true in both happy and unhappy linear counters. So, if we sum up the percentages we get that we have at least 100%, but in general it will be higher due to the fact that it's common to post more than one tweet.

```
linear_counter_happy = ...
linear_counter_morning = ...
...
count = 0
for i in range(0, size(linear_counter_happy)):
        if linear_counter_happy[i] and linear_counter_morning[i]:
                count += 1
return count
```

2.2 Question 2

In order to spell the 30 favorite words of happy users, we can use the space saving algorithm on the words of each happy tweet.

2.3 Question 3

To find the number of distinct words used by the happy users we can use the HyperLogLog data structure. We can use a Bloom Filter to exclude the words repeated only once: if $hash(word) = 0$ set it to 1, otherwise if it is already set to 1 count the word in the HyperLogLog, because it means that we already saw it and it is repeated more than once.

```
bloom_filter = ...
hyper_loglog = ...
for tweet in tweets:
    for word in tweet.text:
        if bloom_filter[hash(word)] == 0:
            bloom_filter[hash(word)] == 1
        else:
            hyper_loglog.put(word)
```

2.4 Question 4

In order to decide if in general the happy messages are longer or shorter then unhappy messages we can scan the whole tweets list and compute the mean of the length of the messages. A better solution would be to use the median, but, since it requires to have the ordered record of all the tweets, we cannot use it in this situation.