Hands-on 7
Analysis on Twitter dataset

Algorithm Design

Silvio Martinico

Jun 18, 2023

Contents

1 The Problem

Questions on twitter data:

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 exclude words repeated only once
4) Decide if in general happy messages are longer or shorter than unhappy messages

2 Question 1

In order to count the percentage of happy and unhappy users in the different moments of the day, we used the *Linear Counter* data structure. In particular, we used seven linear counters: two counters, M_1 and M_2, for the mood of the users (happy/unhappy), four counters $C_1, \ldots, C_4$ for the various moments of the day and a counter T for estimating the total number of users. All the counters are initialized with zeros.

When we insert a new record with user's name u, with a certain mood m_i in a moment of the day d_i, we apply the hash function h, associated with the linear counters, to the user's name, and set $M_i[h(u)]$, $D_i[h(u)]$ and $T[h(u)]$ to 1. Note that the counters M_i and D_i must use the same hash function and have the same length, while T could have different hash function and size.

When counting the number of users with a certain mood m_i in a specific moment of the day d_i, it is sufficient to iterate over the counters M_i and D_i applying an element-wise AND. Each time this results in a 1, we increase a scalar counter which counts the number of users with that mood in that moment of the day. Then, we estimate the total number of users with the number of 1s in T so that we can compute the percentage of total users which have mood m_i in the moment of the day d_i.

If we sum all the percentages of the 8 possible combinations of moods and moments of the day, the resulting percentage is more than 100%. This is due to collisions: if two different users (u, m_i, d_i) and (v, m_j, d_j) collide, but $m_i \neq m_j$ and $d_i \neq d_j$, these two users result in 4 different counter increments, one for each possible combination of the two moments of the day and the two moods: $(m_i, d_i), (m_i, d_j), (m_j, d_i)$ and (m_j, d_j).

3 Question 2

Spelling the 30 favorite words of happy users could be seen as the *Heavy Hitters* problem.

The proposed solution relies on the *Space-Saving algorithm* and returns the 30 words in the table with the biggest guarantee, i.e. the words with the highest counter's lower-bound. In this way we avoid very imprecise counters' approximations.

With a table of size 210 we obtained an accuracy of 100% over the 20500 different words in the tweets of happy users from the dataset `sample.csv` consisting of 20000 records.

4 Question 3

Finding the number of distinct words used by happy users is a *Cardinality Estimation* problem, which can be solved with a linear counter by just counting the number of 1s in the counter.

In order to exclude the words repeated only once, we used a second linear counter C_2 of the same size and with the same hash function h of the first one C_1. When hashing a word w of a happy user, we check first $C_1[h(w)]$: if it is 0 then we set it to 1; if instead it is already 1, it means that we have already seen this word, so we set $C_2[h(w)]$ to 1. Finally, the number of words repeated more than once is estimated by counting the number of 1s in C_2.

5 Question 4

For answering to the fourth question, we don't need probabilistic data structures. In fact, we can measure the average length of the tweets by just keeping two scalar counters: one for happy users and one for unhappy users.

In order to avoid overflow problems, since the length of a tweet is at most 280 characters, we divided the length of each tweet by 280 and then we added it to the counter. In this way, the counter amounts at most to the total number of tweets (the length in this way is a number between 0 and 1).

The analysis found that, on average, unhappy tweets are slightly longer than happy ones.